

**UNIVERSIDADE ESTADUAL PAULISTA “JÚLIO DE MESQUITA FILHO”
FACULDADE DE ENGENHARIA DE ILHA SOLTEIRA
DEPARTAMENTO PÓS-GRADUAÇÃO EM ENGENHARIA ELÉTRICA**

INARA APARECIDA FERRER SILVA

**APLICAÇÕES DE REDES NEURAIS E NEURO FUZZY EM
ENGENHARIA BIOMÉDICA E AGRONOMIA**

Ilha Solteira - SP
2014



UNIVERSIDADE ESTADUAL PAULISTA
"JÚLIO DE MESQUITA FILHO"
Campus de Ilha Solteira

PROGRAMA DE PÓS-GRADUAÇÃO EM ENGENHARIA ELÉTRICA

**APLICAÇÕES DE REDES NEURAIS E NEURO FUZZY EM
ENGENHARIA BIOMÉDICA E AGRONOMIA**

INARA APARECIDA FERRER SILVA

Tese apresentada à Faculdade de Engenharia
do Campus de Ilha Solteira - UNESP como
parte dos requisitos para obtenção do título de
Doutor em Engenharia Elétrica.

Área de Conhecimento: Automação.

Prof. Dr. Marcelo Carvalho Minhoto Teixeira
Orientador

Ilha Solteira - SP

2014

FICHA CATALOGRÁFICA

Desenvolvido pelo Serviço Técnico de Biblioteca e Documentação

- S586a Silva, Inara Aparecida Ferrer.
Aplicações de redes neurais e neuro fuzzy em engenharia biomédica e agronomia / Inara Aparecida Ferrer Silva. -- Ilha Solteira: [s.n.], 2014
80 f. : il.
- Tese (doutorado) - Universidade Estadual Paulista. Faculdade de Engenharia de Ilha Solteira. Área de conhecimento: Automação, 2014
- Orientador: Marcelo Carvalho Minhoto Teixeira
Inclui bibliografia
1. Sistema de inferência neuro fuzzy adaptativo. 2. Lógica nebulosa. 3. GRNN. 4. MLP. 5. RBF. 6. Hemiplegia.



UNIVERSIDADE ESTADUAL PAULISTA
CAMPUS DE ILHA SOLTEIRA
FACULDADE DE ENGENHARIA DE ILHA SOLTEIRA

CERTIFICADO DE APROVAÇÃO

TÍTULO: Aplicações de redes neurais e neuro fuzzy em engenharia biomédica e agronomia

AUTORA: INARA APARECIDA FERRER SILVA

ORIENTADOR: Prof. Dr. MARCELO CARVALHO M TEIXEIRA

Aprovada como parte das exigências para obtenção do Título de DOUTOR EM ENGENHARIA ELÉTRICA, Área: AUTOMAÇÃO, pela Comissão Examinadora:



Prof. Dr. MARCELO CARVALHO M TEIXEIRA

Departamento de Engenharia Elétrica / Faculdade de Engenharia de Ilha Solteira



Prof. Dr. EDVALDO ASSUNÇÃO

Departamento de Engenharia Elétrica / Faculdade de Engenharia de Ilha Solteira



Prof. Dr. APARECIDO AUGUSTO DE CARVALHO

Departamento de Engenharia Elétrica / Faculdade de Engenharia de Ilha Solteira



Prof. Dr. CRISTIANO QUEVEDO ANDREA

Faculdade de Engenharia, Arquitetura e Urbanismo e Geografia / Universidade Federal de Mato Grosso do Sul



Prof. Dr. VALTEMIR EMERENCIO DO NASCIMENTO

Departamento de Área de Informática / Instituto Federal de Educação Ciência e Tecnologia de Mato Grosso

Data da realização: 28 de fevereiro de 2014.

DEDICO

Aos meus pais, **Guido e Irani**, ao meu esposo **Aldo**
e meus amados filhos **Giovanni e Enzo**.

AGRADECIMENTOS

Agradeço, primeiramente, à Deus, por me amparar em todos os momentos difíceis e me fazer forte para finalizar este trabalho.

Ao meu esposo Aldo e meus amados filhos Giovanni e Enzo, que exercitaram a paciência durante a execução deste trabalho.

Aos meus pais, Irani e Guido, pelo amor incondicional e apoio fundamental neste caminho trilhado de minha vida, principalmente durante as viagens de estudo a Ilha Solteira quando ficavam com meus filhos.

A minha irmã Ivana e meus sobrinhos Ian e Iago, pelas palavras amigas para enfrentar esta jornada.

Aos meus sogros Maria Aparecida e José Antônio, cunhados e minha comadre Lúcia pelo apoio durante as viagens e orações de fé para que eu pudesse terminar.

Aos meus familiares que torceram por esta vitória.

Aos docentes, pela dedicação, e técnicos administrativos do programa de pós-graduação em Engenharia Elétrica pelo suporte dos serviços prestados e também aos técnicos da biblioteca, principalmente, Sandra, João, Márcia, Graciele, Rafael e Deoclécio.

Aos colegas e amigos do Departamento de Informática do Instituto Federal de Mato Grosso pelo apoio para a conclusão deste trabalho.

Aos alunos do IFMT que trabalharam com responsabilidade durante a minha ausência.

Ao Instituto Federal de Mato Grosso por proporcionar a realização deste Doutorado.

Aos Professores do DPPG – Departamento de pós-graduação do IFMT pelo suporte neste trabalho de conclusão e equipe de técnicos administrativos.

Ao amigo Marcelo Carvalho Minhoto Teixeira Filho pelo apoio na conclusão deste trabalho.

A amiga Vera Lúcia Macri Gonçalves Teixeira pela generosidade e esforço em nos ajudar durante estes anos.

Ao Prof. Aparecido pela dedicação nas correções, sugestões dos trabalhos, e concessão dos dados utilizados em parte deste trabalho.

Ao Prof. Edvaldo Assunção pela dedicação durante suas aulas, empenho na aquisição do *software* utilizado neste trabalho e sugestões no mesmo.

Ao meu amigo e mestre, Prof. Dr. Marcelo Carvalho Minhoto Teixeira, pelos ensinamentos, paciência e exemplo de dedicação durante estes anos de trabalho. Deixo aqui minha gratidão pelos ensinamentos e profissionalismo para a realização deste trabalho.

O SENHOR é o meu pastor, nada me faltará.
Deitar-me faz em verdes pastos, guia-me mansamente a águas tranquilas. Refrigera a minha alma; guia-me pelas veredas da justiça, por amor do seu nome. Ainda que eu andasse pelo vale da sombra da morte, não temeria mal algum, porque tu estás comigo; a tua vara e o teu cajado me consolam. Preparas uma mesa perante mim na presença dos meus inimigos, unges a minha cabeça com óleo, o meu cálice transborda. Certamente que a bondade e a misericórdia me seguirão todos os dias da minha vida; e habitarei na casa do Senhor por longos dias.
Salmos 23:1-6

RESUMO

Os sistemas *fuzzy* e *neuro fuzzy* têm sido usados com sucesso para resolver problemas em diversas áreas, como medicina, indústria, controle, agronomia e aplicações acadêmicas. Nas últimas décadas, as redes neurais têm sido utilizadas para identificação, avaliação e previsão e dados na medicina e na agronomia. Nesta tese, realizou-se um novo estudo comparativo entre as redes *neuro fuzzy* (ANFIS), rede perceptron multicamadas (MLP), rede função de base radial (RBF) e regressão generalizada (GRNN) na área de engenharia biomédica. Na engenharia biomédica as redes neurais e *neuro fuzzy* foram treinadas e validadas com dados de pacientes hígidos e hemiplégicos (pacientes com sequela motora após acidente vascular cerebral no hemicorpo direito ou esquerdo do cérebro) coletados por meio de um baropodômetro eletrônico (91 indivíduos, sendo 81 hígidos e 10 hemiplégicos). A rede GRNN apresentou o menor erro RMSE (Raiz Quadrada do Erro Médio Quadrático), porém a rede MLP conseguiu identificar um caso de hemiplegia. Na área de agricultura foi proposto um novo estudo comparativo utilizando redes neurais para previsão de produção de trigo (*Triticum aestivum*). Para este estudo utilizou-se uma base de dados experimental de trigo avaliada no período dois anos na região de Selvíria-MS. A validação foi realizada comparando-se a produção estimada pelas redes neurais MLP, GRNN e RBF com a curva de regressão quadrática, comumente utilizada para este fim, e com a rede *neuro fuzzy* ANFIS. O erro RMSE calculado com as redes neurais GRNN e RBF foi menor do que o obtido com a regressão quadrática e com o ANFIS utilizando o treinamento (híbrido). Para validação dos resultados obtidos em hemiplegia utilizou-se o RMSE, a matriz de confusão, a sensibilidade, a especificidade e a acurácia. Os resultados mostraram que a utilização das redes neurais e redes *neuro fuzzy*, na engenharia biomédica, pode ser uma alternativa viável para acompanhamento da evolução do tratamento de pacientes e descoberta de novas informações através da combinação de parâmetros. Na agricultura a metodologia proposta pode trazer benefícios na combinação de vários parâmetros de avaliação de produção, visando otimizar a produção e minimizar custos financeiro em novas plantações.

Palavras-chave: Sistema de inferência *neuro fuzzy* adaptativo (ANFIS). Lógica nebulosa. GRNN. MLP. RBF. Nitrogênio. *Triticum aestivum*. Baropodômetro. Hemiplegia.

ABSTRACT

The fuzzy and neuro fuzzy systems have been successfully used to solve problems in various fields such as medicine, manufacturing, control, agriculture and academic applications. In recent decades, neural networks have been used to the identification, assessment and diagnosis of diseases. In this thesis we performed a comparative study among fuzzy neural networks (ANFIS), multilayer perceptron neural networks (MLP), radial basis function network (RBF) and generalized regression (GRNN) in the area of biomedical engineering and agronomy. In biomedical engineering neural networks and neuro fuzzy were trained and validated with data set from patients (91 subjects, 81 healthy and 10 hemiplegic). The GRNN network had the lowest *Root Mean Square Error* (RMSE), but the MLP network was able to identify a case of hemiplegia. In the area of agriculture a comparative study to estimate the wheat (*Triticum aestivum*) productivity was proposed using neural networks. For this study it was used data from an experimental database of wheat cultivars evaluated during two years in the region of Selvíria - MS. The validation was performed by comparing the estimated productivity through the quadratic regression curve and the output of the ANFIS with the neural networks. The RMSE error calculated with the GRNN and RBF neural networks was lower than that obtained with the quadratic regression and the ANFIS. The results obtained in the study of hemiplegia were validated using the RMSE, the confusion matrix, the sensitivity, the specificity and the error accuracy.

The results showed that the use of neural networks and fuzzy neural networks, in biomedical engineering, can be a viable for monitoring the progress of patients and discovery new information through a combination of parameters.

In agriculture this methodology can bring benefits in combining several evaluation parameters of production to optimize production while minimize financial costs in new plantations.

Keywords: Adaptive neuro fuzzy inference system (ANFIS). Fuzzy logic. GRNN. MLP. RBF. Nitrogen. *Triticum aestivum*. Force plataform. Hemiplegia.

LISTA DE FIGURAS

Figura 1 – Estrutura de um sistema de inferência <i>fuzzy</i>	22
Figura 2 - Arquitetura do ANFIS.	26
Figura 3 - Configuração de neurônio utilizado na derivação do algoritmo <i>backpropagation</i>	33
Figura 4 - Arquitetura da RBF.....	35
Figura 5 - Arquitetura da rede neural GRNN.	36
Figura 6 - Treinamento no ANFIS na classificação de hemiplegia.....	47
Figura 7 - Validação no ANFIS na classificação de hemiplegia.	47
Figura 8 - Validação da rede neural GRNN na classificação de hemiplegia.	48
Figura 9 - Validação da rede neural MLP na classificação de hemiplegia.....	50
Figura 10 - Validação da rede neural RBF na classificação de hemiplegia.	50
Figura 11 - Rede neural RBF na produção de trigo ano 2006.	55
Figura 12 - Rede neural RBF na produção de trigo ano 2007.	56
Figura 13 - Rede neural GRNN na produção de trigo ano 2006.	56
Figura 14 - Rede neural GRNN na produção de trigo ano 2007.	57
Figura 15 - Rede neural MLP na produção de trigo ano 2006.	58
Figura 16 - Rede neural MLP na produção de trigo ano 2007.	58
Figura 17 – Medidas dos sensores do pé esquerdo.....	70
Figura 18 – Medidas dos sensores do pé direito.....	71
Figura 19 – Linguagem de Script MATLAB para a rede MLP com treinamento Levenberg-Marquardt.	72
Figura 20 - Linguagem de Script MATLAB para a rede GRNN.	73
Figura 21 - Linguagem de Script MATLAB para a rede RBF.	74
Figura 22 - Linguagem de script do MATLAB da rede neural RBF para estimativa de produção de trigo ano 2006.	75
Figura 23 - Linguagem de script do MATLAB da rede neural RBF para estimativa de produção de trigo ano 2007.	76

Figura 24 - Linguagem de <i>script</i> do MATLAB da rede neural GRNN para estimativa de produção de trigo ano 2006.	77
Figura 25 - Linguagem de <i>script</i> do MATLAB da rede neural GRNN para estimativa de produção de trigo ano 2007.	78
Figura 26 - Linguagem de <i>script</i> do MATLAB da rede neural MLP para estimativa de produção de trigo ano 2006.	79
Figura 27 - Linguagem de <i>script</i> do MATLAB da rede neural MLP para estimativa de produção de trigo ano 2007.	80

LISTA DE TABELAS

Tabela 1 - Exemplo de matriz de confusão em hemiplegia (diagnóstico).	39
Tabela 2 - Parâmetro ANFIS para o estudo da hemiplegia.	44
Tabela 3 - Conjunto de dados de treinamento e validação.	46
Tabela 4 - Matriz de Confusão do ANFIS na classificação de hemiplegia.	49
Tabela 5 - Matriz de Confusão da rede MLP.	49
Tabela 6 - Matriz de Confusão da rede RBF.	49
Tabela 7 - Matriz de Confusão da rede GRNN.	49
Tabela 8 - Performance de predição das redes neurais.	51
Tabela 9 - Comparação do erro RMSE das redes neurais.	51
Tabela 10 - Dados de treinamento do ano de 2006, valores médios de produção de trigo por dose.	53
Tabela 11 - Dados de treinamento do ano de 2007, valores médios de produção de trigo por dose.	53
Tabela 12 - Estrutura das redes neurais.	54
Tabela 13 - Valor comparativo do RMSE nos modelos de regressão quadrática e ANFIS.	59
Tabela 14 - Valores de produção no ANFIS, regressão quadrática e valores médios de produção real por dose.	59
Tabela 15 - Valor comparativo do RMSE nos modelos de regressão quadrática e ANFIS.	59
Tabela 16 - Estimativa de produção média de grãos de trigo com redes neurais.	60

LISTA DE QUADROS

Quadro 1 - Especificação do sistema de inferência <i>fuzzy</i> no estudo sobre hemiplegia.	43
Quadro 2 - Resumo das estruturas das redes neurais em hemiplegia.	46
Quadro 3 - Arquitetura das redes neurais na produção média de trigo.	54

LISTA DE ABREVIATURAS

ANFIS	Adaptative <i>Neuro Fuzzy</i> Inference System
AVC	Acidente Vascular Cerebral
Max	Máximo
MLP	Multilayer Perceptron
mf	membership function
RBF	Redes de Função de Base Radial
RMSE	Root Mean Square Error
GUI	Graphical User Interface
FIS	<i>Fuzzy</i> Inference System
TS	Takagi Sugeno
RNA	Rede Neural Artificial
GRNN	Generalized Regression Neural Network
SNF	Sistema Neuro Fuzzy

SUMÁRIO

1 INTRODUÇÃO.....	15
2 SISTEMA DE INFERÊNCIA <i>FUZZY</i> E REDES ADAPTATIVAS <i>NEURO FUZZY</i>	21
2.1 Visão Geral sobre Lógica <i>Fuzzy</i>	21
2.2 Sistema de Inferência <i>Fuzzy</i>	21
2.3 Redes Neurais Adaptativas e Sistema de Inferência <i>Neuro Fuzzy</i>	22
2.4 Partição do espaço de entrada.....	28
2.5 Conclusões Parciais	30
3 REDES NEURAIIS ARTIFICIAIS	32
3.1. Rede Neural Perceptron Multicamadas (MLP)	33
3.1.1. Rede perceptron multicamadas com o algoritmo de Levenberg-Marquardt	34
3.2. Rede Neural Função de Base Radial (RBF)	34
3.3 Rede Neural de Regressão Generalizada (GRNN).....	36
3.4 Conclusões Parciais	38
4 DESENVOLVIMENTO DAS APLICAÇÕES UTILIZANDO ANFIS E REDES NEURAIIS	39
4.1 Métodos de Avaliação de Classificação dos Resultados	39
4.2 Desenvolvimento das Aplicações em Hemiplegia	41
4.2.1 Estudo da hemiplegia com o ANFIS	42
4.2.2 Hemiplegia com redes neurais MLP, RBF e GRNN.....	44
4.2.3 Resultados sobre a classificação de hemiplegia	46
4.3 Estudo comparativo de redes neurais para estimar a produção de trigo em função adubação nitrogenada	52
4.4 Resultados.....	55
4.5 Conclusões parciais	60
5 CONSIDERAÇÕES FINAIS	62
5.1 Conclusões.....	62

5.2 Trabalhos futuros	63
5.3 Publicações	63
REFERÊNCIAS	64
APÊNDICE A	70

1 INTRODUÇÃO

No Brasil, a população continuará crescendo até 2042 quando deverá chegar a 228,4 milhões de pessoas. A partir do ano seguinte, diminuirá gradualmente e ficará em torno de 218,2 milhões em 2060. A evolução demográfica no período de 2000 a 2030 acarretará um significativo envelhecimento da população em todas as unidades da federação (INSTITUTO BRASILEIRO DE GEOGRAFIA E ESTATÍSTICA – IBGE, 2013).

O aumento da vida média da população levou a um aumento crescente de pessoas portadoras de doenças, entre elas as doenças cérebro vasculares (DCV), que constituem a terceira causa de morte no mundo e no Brasil representam a primeira causa de morte (LESSA, 1999).

Conforme Davies (1996) a hemiplegia não é uma doença: é uma seqüela neurológica grave devida a um comprometimento circulatório no cérebro com consequências e comprometimento em vários níveis. O indivíduo adulto, que por algum motivo sofreu uma lesão cerebral, compromete uma determinada área do cérebro. Essa área é responsável por determinadas funções no corpo humano; então, dependendo da área lesada, tem-se comprometimento sensorial, mental e/ou físico.

O termo *Soft Computing* representa a combinação de tecnologias inteligentes, como a lógica *fuzzy*, raciocínio probabilístico, redes neurais artificiais (RNA) e algoritmos genéticos para a resolução de problemas. Cada uma dessas tecnologias proporcionam raciocínio e buscam métodos complementares para resolver, problemas complexos do mundo real (BONISSONE, 1997).

A lógica *fuzzy*, baseada na teoria dos conjuntos *fuzzy*, permite a representação de problemas que tratam de conhecimento impreciso, diferentemente dos conceitos bem definidos da lógica clássica. A teoria dos conjuntos *fuzzy*, lógica nebulosa ou lógica *fuzzy*, concebida por Zadeh (1965), é um conjunto de elementos com graus ou níveis contínuos de pertinência que representam em termos matemáticos a informação imprecisa do mundo real. Quando se raciocina na teoria clássica dos conjuntos tem-se que o conceito de pertinência é bem definido, ou um objeto pertence ou não pertence a um conjunto. Na lógica *fuzzy* um objeto x é caracterizado por uma função de pertinência que designa o grau, com valores no intervalo de 0 a 1, indicando o valor com que este elemento objeto pertence a um conjunto *fuzzy* (SILVA et al., 2012).

Os modelos *fuzzy* de acordo com Zadeh (1965) são metodologias para a descrição de sistemas cuja definição de suas classes encontra-se com características muito próximas umas das outras.

As redes neurais artificiais são metodologias computacionais desenvolvidas com base no sistema nervoso humano. Possuem capacidade de adquirir conhecimento (baseado em experiências anteriores) e podem ser definidas como unidades de processamento, caracterizadas por neurônios artificiais, que são interligados por um grande número de interconexões (sinapses artificiais), sendo representadas aqui por vetores/matrizes de pesos sinápticos (SILVA; SPATTI; FLAUZINO, 2010).

As redes neurais artificiais (RNA) são ferramentas computacionais para a classificação de padrões que têm sido o objeto de interesse em pesquisas durante os últimos 15 anos, (KOHAVI; PROVOST, 1998; OHNO-MACHADO; MARK, 1997; PETROSIAN et al., 2001; SHEPPARD et al., 1999). Têm sido utilizadas com sucesso em uma variedade de aplicações médicas (DOMÍNGUEZ HERNÁNDEZ et al., 2013; FREITAS, 2011; RAJESWARI; VAITHYANATHAN; NEELAKANTAN, 2012; YOSHIDA; KAWAGUCHI; TSURUYA, 2013).

Várias ferramentas e metodologias têm sido propostas pelos pesquisadores para o desenvolvimento de sistemas de apoio à decisão médica (GAINO, 2009; GAINO et al., 2011; HOSHI et al., 2005; UBEYLI, 2009; GULER; UBEYLI, 2005). Estas metodologias mostraram-se promissoras para a resolução de problemas antes resolvidos através de técnicas tradicionais.

Ruano e Ruano (2013) usaram a rede neural como um classificador, modelo dinâmico e ferramenta de diagnóstico. Os exemplos apresentados incluíram classificação de fluxo de sangue em êmbolos, com base em sinais de ultrassom transcraniano, modelagem de temperatura do tecido baseada em dados, e identificação de áreas de acidente vascular cerebral isquêmico baseada em imagens de tomografia em computador. As redes neurais produziram resultados muito precisos.

Wu et al. (2010) verificaram a possibilidade de utilizar a Rede de Função de Base Radial (RBF), para reconhecer de forma precisa e prever o início dos tremores da doença de Parkinson em seres humanos. Os dados relativos à formação RBF foram obtidos por meio de eletrodos implantados no paciente com a doença de Parkinson. A eficácia das RBF foi inicialmente demonstrada por um estudo de caso real.

Pan et al. (2012) realizaram um estudo comparativo entre Máquinas de Vetores de Suporte (SVM) e redes neurais (Perceptron Multicamadas e Redes de Função de Base Radial) para a classificação do tremor da doença de Parkinson. Seus resultados mostraram que máquinas de vetores e suporte obtiveram resultados superiores às redes neurais.

Das, Turkoglu e Sengur (2009) introduziram uma metodologia que usa *SAS software* 9.1.3 para o diagnóstico de doenças no coração. Eles obtiveram uma precisão de classificação de 89,01% nas experiências feitas utilizando dados obtidos a partir do banco de dados de doença cardíaca. Eles também obtiveram 80,95% de desempenho nos valores de sensibilidade e 95,91% de especificidade, no diagnóstico de doenças do coração.

Temurtas (2009) realizou um estudo comparativo no diagnóstico da doença da tireoide usando redes neurais multicamadas, probabilística e vetor de quantização (LVQ-Learning vector quantization). Os resultados mostraram que a rede neural probabilística obteve a melhor acurácia de classificação para a base de dados da tireoide.

Um problema importante em aplicações médicas é o uso de eletroestimulação para movimentar músculos em pacientes paraplégicos (VANONCINI et al., 2012). Estudos e simulações de controle de posição da perna de um paraplégico com estimulação de sinais elétricos foram realizados, usando modelos *fuzzy* Takagi-Sugeno (TS) (GAINO, 2009; GAINO et al., 2011; TEIXEIRA; ASSUNÇÃO; AVELLAR, 2003; TEIXEIRA et al., 2006; TEIXEIRA; ZAK, 1999).

Freitas (2008) implementou um baropodômetro eletrônico para analisar a distribuição de pesos na região plantar dos pacientes e analisar a postura dos indivíduos. O sistema é composto por duas plataformas de força que utilizam condicionamento moderno e os circuitos de sinais de interface. Cada plataforma tem 24 células de carga, construídas com extensômetros. O equipamento possui resposta linear com coeficiente de determinação de 0,9997, baixa histerese, precisão de 0,84% e resolução de 0,5 N. As medidas são exibidas na tela de um computador. Usando este baropodômetro, foram avaliados 81 indivíduos hígidos e 7 hemiplégicos.

Segundo Freitas (2011), o baropodômetro é um equipamento, desenvolvido para medir as pressões exercidas pelos pés e suas influências no caminhar e na postura corporal das pessoas. Os exames são realizados através de uma plataforma fixa e de uma palmilha baropodométrica, especialmente dotados de centenas de microsensores, que fazem os registros dinâmico e estático. Os dados são armazenados em softwares especiais que avaliam a pressão em cada ponto do pé e os seus respectivos picos de força, além de acompanhar a

dinâmica completa do passo (contato, apoio, propulsão, velocidade e distância) e os desvios e equilíbrio posturais.

Freitas (2011) utilizou os dados obtidos com este baropodômetro, proposto por Freitas (2008), para realizar a previsão de hemiplegia utilizando a rede neural ARTMAP nebulosa e a rede MLP com *backpropagation*. Os resultados mostraram que a rede ARTMAP nebulosa classificou com melhor precisão os casos de hemiplegia do que a MLP.

Chen et al. (2004) realizaram um estudo visando obter uma rede neural e um controle *fuzzy* para um sistema de estimulação elétrica (FES) usado para ajustar a eletroestimulação ideal para controlar o movimento de articulação do tornozelo em hemiplégicos. Avaliaram-se quatro indivíduos no teste clínico. Verificou-se que tanto o erro quadrático médio quanto o erro médio foram mínimos quando se utiliza a rede neural e controlador *fuzzy*. O problema na locomoção do hemiplégico foi efetivamente melhorado incorporando a rede neural e o controlador *fuzzy* com o simulador elétrico funcional.

Lau, Tong e Zhu (2009) pesquisaram o uso de máquina de vetor de suporte (SVM) para classificar as diferentes condições de caminhada para hemiplégicos. Sete participantes em cinco condições diferentes: nível horizontal, subir escada, descer escada, curva ascendente e curva descendente. O desempenho da classificação de subir escadas, descer escadas e outras condições de caminhada foi de 92,9 %, utilizando recursos de entrada do sensor. Foi ainda melhorada para 97,5 % pela adição de mais duas entradas dos sensores ligados ao pé. Subir escada também foi classificada pelas entradas do sensor do pé com uma precisão de 96%. O desempenho de um SVM mostrou-se superior à de outros métodos de aprendizado de máquina utilizando redes neurais artificiais (RNA) e rede neural função de base radial (RBF).

Nos últimos anos, intensificou-se a aplicação das redes neurais na modelagem e predição de dados na área agrônômica.

As redes neurais foram utilizadas para avaliação da produtividade do solo, buscando a classificação de terras para irrigação, no estudo realizado por (BUCENE; RODRIGUES, 2004).

Em Benini (2007) utilizou-se o sistema de inferência *neuro fuzzy* para estimar a densidade dos solos. Seus resultados mostraram que o sistema *neuro fuzzy* (SNF) pode ser utilizado para a estimativa da densidade do solo com eficiência, que esta metodologia gera modelos com boa capacidade de generalização, e que pode ser implementada utilizando dados laboratoriais.

Estudos de modelagem utilizando redes neurais com a cultura do trigo foram observados em (KASHANINEJAD; DEHGHANI; KASHIRI, 2009; ALVAREZ, 2009).

Em Silva et al. (2014) propôs-se uma metodologia para estimação da produção de trigo em função da adubação nitrogenada utilizando as redes *neuro fuzzy*. Os resultados mostraram que a metodologia foi viável, pois o erro RMSE da produção estimada pela rede *neuro fuzzy*, foi menor que o RMSE da curva de regressão quadrática, comumente utilizada para este fim.

Um sistema de inferência *neuro fuzzy* adaptativo (ANFIS) foi implementado para avaliar diferentes combinações de fluxo (no bico injetor) e altura da barra de lançamento, na uniformidade de distribuição de pesticida líquido de um pulverizador de campo. Além disso, o ANFIS foi utilizado para determinar a melhor combinação dos dois fatores principais (altura de lançamento e taxa de fluxo do bico injetor) que resultariam em melhor uniformidade de distribuição. No ANFIS, os dois fatores principais foram selecionados como entradas, no entanto, o Coeficiente de Uniformidade de Distribuição (CDU) foi considerado como a saída do sistema. Os resultados do estudo mostraram que a técnica ANFIS foi eficaz em avaliar e classificar as diferentes combinações possíveis dos fatores principais envolvidos para melhor uniformidade de distribuição. Além disso, os resultados revelaram que a ANFIS utilizado foi preciso em prever a CDU (AL-GAADI; ABOUKARIMA; SAYEDAHMED, 2011).

Segundo estatísticas, há 24,5 milhões de portadores de deficiências no Brasil. Muitos têm uma vida ativa e precisam de exercícios de reabilitação visando a reversão do problema (DAVIES, 1996).

Os pacientes acometidos por uma doença vascular cerebral apresentam frequentemente deficiência na mobilidade motora, na linguagem, no aprendizado e na memória. A deficiência na mobilidade motora determina uma sequela denominada hemiplegia que consiste em um estado físico caracterizado por uma paralisia de um hemicorpo (um lado do cérebro), levando à incapacidade ou dificuldade em realizar diversas tarefas da vida diária que podem interromper atividades de extrema importância na realização pessoal (LESSA, 1999).

Os experimentos na área agrícola envolvem custos de mão de obra, sementes, tempo de observação e avaliação, local para plantação e outros fatores, que provocam elevado custo financeiro, e muitas vezes, inviabilizam a implementação de pesquisas nesta área.

As redes neurais são metodologias que possuem a habilidade em reconhecer (aprender) de forma precisa e prever, através de casos reais observados, novos casos. Podem ser aplicadas em diversas áreas da medicina.

O sistema de inferência *fuzzy* é um sistema híbrido, resultado da combinação da metodologia da lógica *fuzzy* com uma rede neural adaptativa. Usando um conjunto de dados de entrada/saída realiza a modelagem *fuzzy*, mapeando informações sobre este conjunto de dados (SILVA; SILVA; TEIXEIRA, 2012).

Através dos exemplos de aplicação, esta tese propõe o uso das técnicas inteligentes, tais como as redes neurais e o sistema de inferência *neuro fuzzy* visando a busca de soluções, de baixo custo, na avaliação e identificação de casos em engenharia biomédica e agronomia que promovam avanços nas pesquisas para a área de engenharia biomédica e na agronomia.

Neste estudo, avaliou-se o desempenho de vários classificadores neurais e redes *neuro fuzzy* e suas capacidades de previsão. Os objetivos específicos deste trabalho foram:

1. Realizar um estudo comparativo das redes neurais e rede adaptativa *neuro fuzzy* (ANFIS) para a previsão de casos de hemiplegia baseado em dados obtidos através de um baropodômetro (plataforma de força);
2. Utilizar as redes neurais, através de estudo comparativo, para realizar a estimativa da produção de trigo utilizando doses de nitrogênio como entradas do sistema.

O trabalho foi organizado em cinco capítulos como segue:

Capítulo 1: Introdução do trabalho que mostra os objetivos da tese justifica a aplicação de técnicas inteligentes (*soft computing*) nas áreas de engenharia biomédica e agricultura, e finalmente apresenta a organização do trabalho.

Capítulo 2: Aborda os conceitos e desenvolvimento de redes neurais nebulosas ou redes *neuro fuzzy* e suas aplicações. Descrevem-se os sistemas *fuzzy* e *neuro fuzzy* seus principais conceitos, destacando os métodos de inferência e classificação.

Capítulo 3: Aborda conceitos relacionados às redes neurais RBF, GRNN e MLP.

Capítulo 4: Apresenta o estudo comparativo das redes neurais e redes *neuro fuzzy* na identificação de hemiplegia e um exemplo de aplicação utilizando redes neurais para estimativa da produção média de grãos de trigo.

Capítulo 5: Apresenta as conclusões obtidas através deste trabalho, sugestões de trabalhos futuros e publicações em congressos e periódicos.

2 SISTEMA DE INFERÊNCIA *FUZZY* E REDES ADAPTATIVAS *NEURO FUZZY*

Neste capítulo são apresentados conceitos sobre lógica *fuzzy* e redes adaptativas *neuro fuzzy* bem como sua aplicação em sistemas do mundo real através dos exemplos apresentados nesta tese.

2.1 Visão Geral sobre Lógica *Fuzzy*

A teoria dos conjuntos nebulosos ou teoria dos conjuntos *fuzzy*, foi desenvolvida por Zadeh (1965), através de seus trabalhos como professor na Universidade da Califórnia em Berkeley. A teoria dos conjuntos *fuzzy* ou lógica *fuzzy*, como também é conhecida, aborda uma classe de elementos com graus ou níveis contínuos de pertinência que traduzem em termos matemáticos a informação imprecisa descrita por um conjunto de regras do mundo real. A função que define o grau de pertinência de um determinado elemento em um conjunto *fuzzy*, levando em consideração o seu universo de discurso, é definida como função de pertinência. A função de pertinência $\mu_A(x)$ indica o grau de compatibilidade entre x e o conceito expresso por A .

O conceito da lógica *fuzzy* é amplamente entendido no mundo real, no qual expressões imprecisas e ambíguas são comumente utilizadas, tais como “muito”, “pouco”, “alto”, “baixo”, “jovem”, “adulto”, e assimiladas pelo raciocínio humano (TANSCHKEIT; GOMIDE; TEIXEIRA, 2007; ZADEH, 1968; ZADEH, 1973).

A modelagem *fuzzy* é um método que permite descrever as características de um sistema usando regras de inferência *fuzzy* (TAKAGI; SUGENO, 1985).

2.2 Sistema de Inferência *Fuzzy*

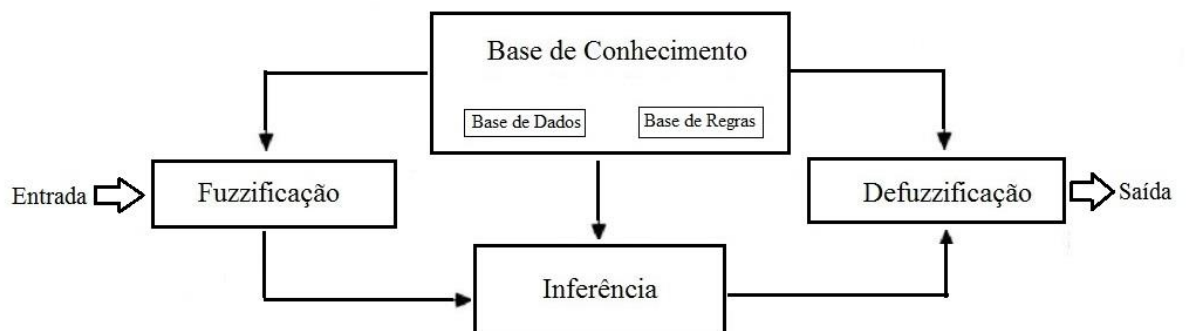
Os sistemas de inferência *fuzzy* (FIS) são também conhecidos como sistemas baseados em regras, modelos *fuzzy*, memórias associativas *fuzzy* (FAM), ou controladores *fuzzy* quando usados em controle (JANG, 1995).

Um dos maiores problemas dos FIS é a definição dos valores das variáveis linguísticas e das regras, uma vez que há necessidade de um especialista representar este conhecimento.

Basicamente o modelo de um sistema de inferência *fuzzy* é composto por 5 partes (JANG, 1993), descritas na Figura 1:

1. Interface de fuzzificação que transforma as informações de entrada em conjuntos *fuzzy*;
2. A base de regras que contém as regras do tipo “se ... então” elaboradas pelos especialistas;
3. A base de dados que contém as funções de pertinência dos conjuntos *fuzzy* usados nas regras. Normalmente a base de dados e a base de regras são unificadas formando a base de conhecimento.
4. O método de inferência para saída *fuzzy*;
5. Uma interface de defuzzificação, cuja função é converter os valores *fuzzy* em valores precisos.

Figura 1 – Estrutura de um sistema de inferência *fuzzy*.



Fonte: Jang (1993).

2.3 Redes Neurais Adaptativas e Sistema de Inferência *Neuro Fuzzy*

Rede adaptativa é uma rede multicamadas, com arquitetura não recorrente (*feedforward*) disposta por nós interligados por conexões unidirecionais e aprendizado supervisionado. Cada nó corresponde a uma unidade de processamento, que recebe dados, processam esses dados através de uma função particular e emite uma saída (JANG, 1993).

Através do conjunto de dados, que corresponde ao vetor de entrada, a rede adaptativa tem como objetivo ajustar os pesos dos nós de forma a minimizar a diferença entre a saída da rede e a saída desejada.

O nome dado a este ajuste de pesos, realizado pela rede adaptativa, é treinamento. Cada treinamento utiliza um algoritmo de otimização para ajustar os pesos e minimizar o erro de saída da rede. O treinamento comum nestas redes é baseado no gradiente descendente proposto por Werbos (1974) ou treinamento híbrido proposto por Jang (1993).

Os sistemas híbridos são compostos por duas ou mais técnicas de modelagem, visando a obtenção de um sistema com maior capacidade de interpretação, aprendizado, estimativa e generalização (SILVA; SPATTI; FLAUZINO, 2010).

Babuska e Verbruggen (2003) afirmam que os sistemas híbridos *neuro fuzzy* combinam a semântica transparente dos sistemas *fuzzy* baseados em regras com a capacidade de aprendizado das redes neurais artificiais.

Um dos primeiros sistemas híbridos *neuro fuzzy* foi o ANFIS (Adaptive *Neuro Fuzzy* Inference System), que a partir dos vetores de entradas, do tipo e número das funções de pertinência das entradas e do vetor de saída, consegue, por intermédio dos algoritmos *backpropagation* e híbrido, ajustar as funções de pertinência das entradas, otimizando a fidelidade da inferência sobre a planta representada (SILVA; SPATTI; FLAUZINO, 2010).

O ANFIS, desenvolvido por Jang (1993), é considerado um sistema de inferência *fuzzy* organizado na forma de uma rede adaptativa, capaz de fazer o mapeamento dos dados de entrada e saída com base no conhecimento de um especialista.

As principais motivações de Jang (1993) para este desenvolvimento foram:

1. Inexistência de técnicas para transformar o conhecimento humano em regras ou base de dados *fuzzy* com funções de pertinência apropriadas;
2. Ausência de métodos que permitissem otimizar funções de pertinência para minimizar o erro ou melhorar a performance.

No ANFIS o conhecimento do especialista é utilizado para definição das variáveis de entrada e saída, na definição do número de termos linguísticos, e nos parâmetros iniciais das funções de pertinência. O número de regras é definido em função do número de funções de pertinência de cada variável quando a arquitetura deste sistema é rígida.

Existem vários tipos de Sistema de Inferência *Fuzzy* (FIS) na literatura. Dependendo do raciocínio e das regras aplicadas, muitos FIS podem ser classificados em três tipos, Jang (1993). Exemplo:

1. Tipo 1- FIS Tsukamoto: O resultado final é a média ponderada de saída *crisp* (real) de cada regra induzida por pesos (o produto ou mínimo dos graus de correspondência com a parte antecedente) e funções de pertinência de saída. As

funções de pertinência de saída utilizadas neste esquema devem ser funções monótonas (monotônica). As regras são do tipo “Se x é A e y é B então $z = C$ ”;

2. Tipo 2 - FIS Mamdani: As regras são do tipo “Se x é A e y é B então $z = C$ ” (A , B , C são conjuntos *fuzzy*). O resultado é obtido pela aplicação da operação “max” às saídas *fuzzy*. Vários esquemas têm sido propostos para escolher a saída real (crisp) final com base na saída *fuzzy*, alguns deles são centróide da área, bissetriz de área, com média de máxima, critério máximo, etc;
3. Tipo 3 - FIS Takagi-Sugeno: As regras são do tipo “Se x é A e y é B então $z = f(x,y)$ ”. O consequente da regra é uma combinação linear (função não *fuzzy*) das variáveis de entrada somadas ao termo constante. A saída final é a média ponderada de cada regra de saída.

Em um conjunto clássico os elementos podem ou não pertencer a ele. Um conjunto *fuzzy* expressa o grau com que um elemento pertence a um conjunto. As funções características de um conjunto *fuzzy* permitem ter valores entre 0 e 1, o qual denota o grau de pertinência do elemento no conjunto dado. Então um conjunto *fuzzy* A em X é definido como um conjunto ordenado de pares: $A = \{(x, \mu_A(x)) / x \in X\}$. $\mu_A(x)$ é chamada de função de pertinência (mf) para o conjunto *fuzzy* A . A mf mapeia cada elemento x com um grau de pertinência ou valor entre 0 e 1 (ZADEH, 1968).

O modelo de inferência *fuzzy* Takagi e Sugeno (1985) representa um sistema que associa um conjunto de regras linguísticas no antecedente (parte “se”) com proposições *fuzzy*, e no consequente (parte “então”) são apresentadas expressões funcionais, do tipo $y=f(x)$, das variáveis linguísticas do antecedente. Cada regra deste modelo representa um sistema linear local somando a uma constante, enquanto o modelo por si só descreve um sistema não linear.

Cada regra do modelo *fuzzy* tem a seguinte forma:

Se x_1 é A_1^i e x_2 é A_2^i ...e x_m é A_m^i então

Regra i:

$$y^i = \theta_0^i + \sum_{j=1}^m \theta_j^i x_j,$$

sendo que:

- $i = 1, 2, \dots, m$ (m é o número de regras);
- x_1, x_2, \dots, x_m são as variáveis de entrada;

- $A_1^i, A_2^i, \dots, A_m^i$ são conjuntos *fuzzy* no antecedente das regras de um sistema com múltiplas entradas e uma única saída, nos universos de discursos U_1, U_2, \dots, U_m ;
- $y^i = f_i(x)$, $i = 1, 2, \dots, m$, são funções lineares ou não lineares das variáveis numéricas de entrada;
- $\theta_0^i \dots \theta_j^i$ são parâmetros consequentes das regras.

Definindo $x_k = [x_{k1}, x_{k2}, \dots, x_{km}]^T$ ($k = 1, 2, \dots, n$) e $y_k = f(x) = f(x_1, x_2, \dots, x_n)$ sendo que os n conjuntos *fuzzy* $A_1^j, A_2^j, \dots, A_n^j$ do antecedente da regra i ($i = 1, 2, \dots, m$) formam uma região $A_1^j \times A_2^j \times \dots \times A_n^j$ no espaço de entrada de x_1, x_2, \dots, x_m , respectivamente denominada superfície *fuzzy* de entrada-saída.

A saída do modelo Takagi e Sugeno (1985) é definida como a soma ponderada dos consequentes y_i , ($i = 1, 2, \dots, m$), de cada subsistema afim, dada por:

$$y = \sum_{i=1}^m \tau_i y_i, \quad (1)$$

$$\tau_i(x) = \frac{h_i(x)}{\sum_{j=1}^m h_j}. \quad (2)$$

A Equação (2) representa o grau de ativação (grau de pertinência) normalizado para a regra i ($i = 1, 2, \dots, m$), com grau de ativação h_i , para a regra i dada por:

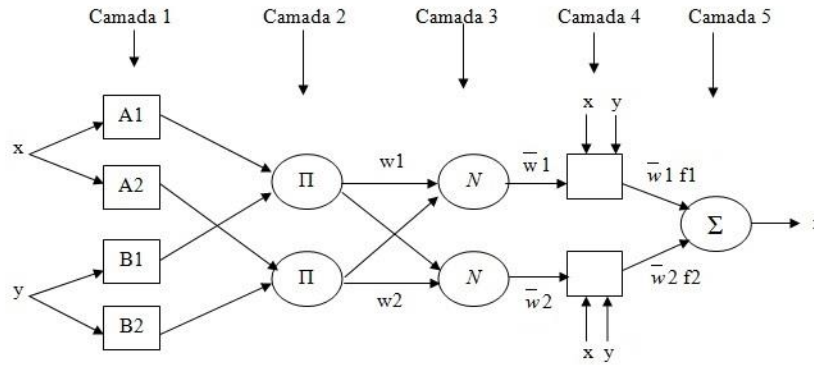
$$h_i(x) = \mu_{A_{1,i}}(x_1) \wedge \mu_{A_{2,i}}(x_2) \wedge \dots \wedge \mu_{A_{n,i}}(x_n), \quad (3)$$

sendo que, x_j ($j = 1, 2, \dots, m$) é um ponto do universo de discurso U_j ($j = 1, 2, \dots, n$).

Um dos sistemas *neuro fuzzy* mais utilizados é o ANFIS, empregado em aplicações de previsão e aproximação de funções nas quais, a partir de um conjunto de dados para treinamento (pares de entrada e saída), e por intermédio do sistema de inferência Takagi-Sugeno, geram regras *fuzzy* (JANG, 1993), conforme ilustrado na Figura 2.

Para fins didáticos imagina-se uma estrutura da rede ANFIS para a situação de duas entradas $\{x,y\}$, e uma saída f . A arquitetura ANFIS (tipo 3-Takagi-Sugeno) é composta por cinco camadas, cada uma com uma finalidade específica (JANG, 1993).

Figura 2 - Arquitetura do ANFIS.



Fonte: Jang (1993).

Camada 1: Calcula-se o grau de pertinência das entradas x e y , de acordo com o tipo de função de pertinência escolhida nestes nós (A1, A2, B1, B2). Normalmente, a função de pertinência dos antecedentes é gaussiana:

$$\mu_{A_i}(x), \mu_{B_i}(y), \quad i = 1, 2. \quad (4)$$

Camada 2: Nesta camada, os neurônios executam a operação de T-norma (norma triangular, representada pelo produtório ou operação mínimo dos conjuntos *fuzzy*), efetuando o produtório (multiplicação) dos sinais de entrada e enviando-os para saída:

$$w_i = \mu_{A_i}(x) \times \mu_{B_i}(y), \quad i = 1, 2. \quad (5)$$

Camada 3: Esta camada normaliza as funções de pertinência. As saídas dos nós desta camada são dadas por:

$$\bar{w}_i = \frac{w_i}{w_1 + w_2}, \quad i = 1, 2. \quad (6)$$

Camada 4: Nesta camada estão os parâmetros consequentes da regra. Os valores de saída de cada nó desta camada são dados por:

$$f_1 = p_1 x + q_1 y + r_1, \quad (7)$$

$$f_2 = p_2 x + q_2 y + r_2, \quad (8)$$

$$\bar{w}_i f_i, i = 1, 2. \quad (9)$$

sendo que os valores de f_1 e f_2 correspondem aos consequentes Takagi-Sugeno de primeira ordem (combinações lineares das entradas somadas a constantes).

Camada 5: O nó desta última camada da arquitetura calcula a saída precisa do sistema e, juntamente com os nós das camadas 3 e 4, promovem a defuzzificação ou soma total de todos os sinais de entrada. Sua saída é dada por:

$$\sum_i \bar{w}_i f_i = \frac{\sum_i w_i f_i}{\sum_i w_i}, i = 1, 2. \quad (10)$$

sendo que f_i é a combinação afim dos parâmetros consequentes p_i , q_i e r_i .

Para representar as diferentes capacidades adaptativas, usa-se tanto círculo e nós quadrados em uma rede adaptativa, Figura 3. Um nó adaptativo (com formato quadrado) tem parâmetros enquanto um nó adaptativo com formato de círculo (nó fixo) não tem nenhum. O conjunto de parâmetros de uma rede adaptativa é a união dos conjuntos de parâmetros de cada nó adaptativo. A fim de alcançar um mapeamento de entrada-saída desejado, estes parâmetros são atualizados de acordo com dados de treinamento e de um procedimento de aprendizagem. Pode-se aplicar o método do gradiente descendente para identificar os parâmetros em uma rede adaptativa, porém o método é geralmente lento e propenso a mínimos locais. Assim foi proposta uma regra de aprendizagem híbrida que combina o método de gradiente descendente nos parâmetros antecedentes e os mínimos quadrados (LSE) para estimar ou identificar parâmetros consequentes, (JANG, 1993).

O algoritmo do modelo *neuro fuzzy* é implementado no ANFIS do MATLAB (MATHWORKS, 1995) da seguinte forma:

- Elaboração do arquivo de treinamento;
- Elaboração do arquivo de validação;
- Carregamento dos arquivos de treinamento e validação;
- Determinação do número e dos tipos de funções de pertinências da entrada e saída no caso de escolher *grid partition* (partição do espaço de entradas). Para a função *subtractive cluster* especificar o raio de influência da base da função gaussiana, fator multiplicador, raio de aceitação no agrupamento e raio de rejeição;

- Escolha do treinamento da rede neural (*backpropagation* ou híbrido). Neste caso utilizou-se o híbrido;
- Escolha do erro (valor zero) e época desejada;
- Início do treinamento da rede neural;
- Teste do modelo *neuro fuzzy* gerado com os dados de treinamento e validação.

O sistema de inferência adaptativo *neuro fuzzy* pode ser implementado através da função *anfisedit* da toolbox *fuzzy logic* do MATLAB (MATHWORKS, 1995) ou executado através da linha de comando do aplicativo criando-se os arquivos no editor do programa.

Sistema *fuzzy* para identificação e modelagem tem sido um assunto de intensa pesquisa em várias décadas (TAKAGI; SUGENO, 1985; TEIXEIRA; ASSUNÇÃO; AVELLAR, 2003; TEIXEIRA et al., 2006; TEIXEIRA; ZAK, 1999; SUGENO; KANG, 1988; WANG; LANGARI, 1996; WANG, 1994).

A modelagem das regras *fuzzy* são um ponto crítico no desenvolvimento de sistemas *fuzzy* porque a aquisição de conhecimento pode ser uma tarefa complexa, quando não há especialistas.

2.4 Partição do espaço de entrada

Os sistemas *neuro fuzzy* realizam, internamente, um mapeamento entre regiões do espaço de entrada em regiões *fuzzy* do espaço de saída, através de regras *fuzzy* do sistema. As regiões *fuzzy* do espaço de entrada são determinadas no processo de identificação da estrutura. Nesse processo, os espaços de entrada e/ou saída são divididos segundo um determinado método de partição. No sistema *neuro fuzzy* utilizado neste trabalho, utilizou-se somente a partição *subtractive cluster* no exemplo de aplicação.

Partição *fuzzy grid*

Segundo Yu e Li (2008) muitas estruturas de identificação de sistemas têm sido propostas. Estas incluem *fuzzy grid*, *clustering on-line* e *off-line* (agrupamento), redes neurais, algoritmos genéticos e máquina vetor de suporte (SVM). *Fuzzy grid* é um método bem conhecido no qual a entrada e saída são particionadas em malhas (*grids*).

A partição *grid* mapeia regiões *fuzzy* de entrada em regiões *fuzzy* de saída através das regras. Estas regiões são criadas na definição das funções de pertinência de entrada.

O antecedente de uma regra *fuzzy* define uma região *fuzzy* no espaço das variáveis de entrada. O consequente descreve o comportamento da variável de saída dentro da região *fuzzy* definida no antecedente.

O uso da partição *grid* limita-se ao número de entradas do sistema, uma vez que o número de regras geradas aumenta exponencialmente com o aumento do número de entradas. Por exemplo, se um sistema possui m entradas e n funções de pertinência para cada entrada, tem-se que o número de regras *fuzzy* total é calculada como n^m regras.

Partição agrupamento subtrativo (*subtractive cluster*)

Agrupamento é uma classificação não supervisionada de padrões (observações, dados, objetos, exemplos) em grupos (*clusters*). Intuitivamente, padrões semelhantes devem estar no mesmo grupo e padrões distintos devem estar em grupos distintos. Agrupamento é uma técnica dirigida por dados, isto é, não se conhece previamente a que grupo os dados pertencem.

Devido a problemas práticos no mundo real, a geração automática de modelos *fuzzy* a partir de conjuntos de dados, surgiu na literatura envolvendo técnicas baseadas em agrupamento (CHEN; XI; ZHANG, 1998), redes neurais (YU; LI, 2008) e algoritmos genéticos (CHOI; PEDRYCZ, 2008; GOMEZ-SKARMETA; DELGADO; VILA, 1999; PAPADAKIS; THEOCHARIS, 2002).

Em algumas situações a identificação de sistemas pode ser feita pela técnica de agrupamento *fuzzy*, o qual agrupa os dados similares e depois traduz estes agrupamentos para regras *fuzzy*.

As técnicas de modelagem baseadas em agrupamento *fuzzy* são utilizadas quando as técnicas tradicionais falham ou não conseguem uma resposta satisfatória para determinados dados de processos não lineares.

A ideia é dividir os dados em agrupamentos *fuzzy*, cada um representando parte do comportamento do sistema.

A identificação de sistemas usando agrupamento *fuzzy* envolve formação de agrupamentos no espaço de estados e tradução destes agrupamentos em regras Takagi Sugeno tal que o modelo obtido seja identificado.

Os métodos de agrupamento *fuzzy* têm sido amplamente aplicados em várias áreas, como processamento de imagem, a recuperação da informação, mineração de dados e outras (JAIN; MURTY; FLYNN, 1999).

Segundo Sivaraman e Arulselvi (2011), a representação dos conjuntos, na forma de vetores de entrada e saída, é utilizada quando não se conhece previamente a dinâmica de um sistema. O agrupamento *fuzzy* é uma técnica que pode ser usada para identificação das partições. Assim cada *cluster* pode ser interpretado como um modelo local linear do sistema.

A técnica de busca com capacidade de gerar automaticamente o número de *cluster* e os seus centros e foi introduzido pelo método da montanha (YAGER; FILEV, 1994). Este método considera cada ponto dos dados como um centro do agrupamento em potencial.

O algoritmo *subtractive cluster* é um melhoramento do método da montanha, um algoritmo não supervisionado, ou seja, não é necessário dizer quais são os centros. Este algoritmo procura os centros baseados na densidade dos pontos da amostra. O ponto com mais alto número de vizinhos é selecionado como o centro do agrupamento. Em seguida este é removido e o algoritmo procura por novos pontos com alto número de vizinhos. Somente os pontos com maior número de vizinhos são candidatos a centro. Com o cálculo da densidade em cada ponto, aquele com densidade mais alta será o centro deste agrupamento (CHIU, 1994).

O algoritmo descrito por Chiu (1994) é utilizado na estimação do número de regras necessárias à definição de um modelo *fuzzy* baseado em um conjunto de dados, e não necessita ser especificado previamente o número de agrupamentos no modelo. Utilizado quando não se conhece o número de agrupamentos de um conjunto de dados, o agrupamento subtrativo é um algoritmo para definir este número.

2.5 Conclusões Parciais

Este capítulo introduz conceitos da lógica *fuzzy*, redes adaptativas e sistemas de inferência *fuzzy*.

A teoria dos conjuntos *fuzzy* é uma metodologia muito importante para lidar com a incerteza. Tem sido utilizada com êxito em diversas aplicações médicas, na indústria, em controle, agronomia e outras.

O ANFIS é um sistema híbrido, pois utiliza a lógica *fuzzy* para a concepção das entradas do sistema, e a capacidade de aprendizagem das redes neurais adaptativas para fazer

a aproximação de funções ou predição de novos dados ou modelagem de um sistema desconhecido.

A lógica *fuzzy* é usada quando se tem um especialista para descrever a base de conhecimento, ou seja, as regras que representam o modelo do sistema. Quando não se conhece o modelo do sistema e não se tem a presença do especialista, através dos dados de entrada e saída do sistema, é possível utilizar o sistema ANFIS para identificar o modelo.

As regras obtidas através do ANFIS são facilmente entendidas e interpretadas.

O ANFIS possui duas alternativas de partição dos espaços de entrada para desenvolvimento do modelo. São elas: a partição *grid* e o agrupamento subtrativo (*subtractive cluster*).

3 REDES NEURAIS ARTIFICIAIS

Redes neurais são metodologias baseadas no funcionamento do cérebro e sistema nervoso. Tradicionalmente têm sido utilizadas para imitar tarefas cognitivas realizadas pelo cérebro humano.

As redes neurais artificiais são modelos computacionais inspirados no sistema nervoso de seres vivos. Possuem capacidade de aquisição do conhecimento (baseado em informações fornecidas pelo ser humano) e podem ser definidas como um conjunto de unidades de processamento, caracterizadas por neurônios artificiais interligados por um grande número de interconexões (sinapses artificiais), sendo representadas por vetores/matrizes de pesos sinápticos (SILVA; SPATTI; FLAUZINO, 2010).

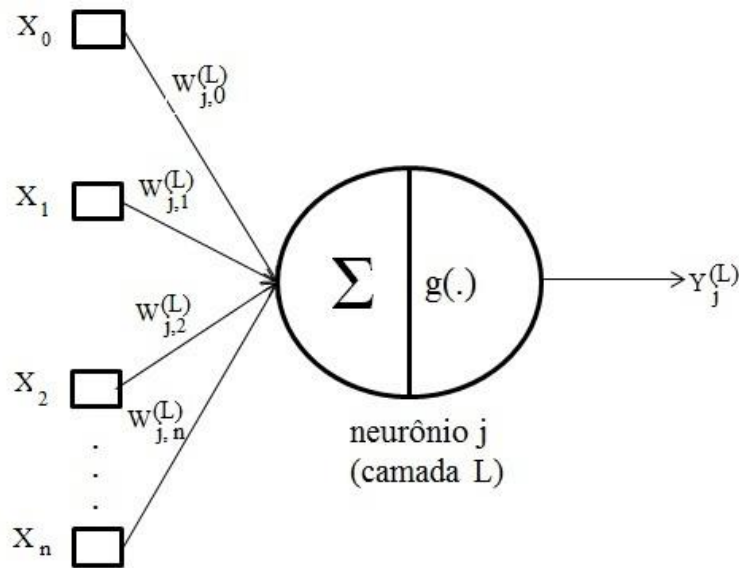
Conforme Silva, Spatti e Flauzino (2010) as redes neurais artificiais podem ser empregadas em diversos problemas relacionados às engenharias e ciências. As áreas de aplicabilidade podem ser enquadradas conforme a seguir:

1. Aproximador universal de funções: o objetivo consiste em mapear o relacionamento funcional entre as variáveis (tipicamente reais) de um sistema a partir de um conjunto conhecido de seus valores representativos. Normalmente envolvem aplicações cujo mapeamento por técnicas convencionais são de difícil obtenção;
2. Controle de processos: consiste em identificar ações de controle que permitam o alcance dos requisitos de qualidade, eficiência e segurança do processo;
3. Reconhecimento/classificação de padrões: associa padrões de entrada (amostra) para uma das classes previamente definidas, como acontece em reconhecimento de imagem, som, voz, escrita;
4. Agrupamento de dados: identificação de similaridades e particularidades entre os diversos padrões de entrada a fim de realizar o agrupamento;
5. Sistema de previsão: estimação de valores futuros baseando-se nas medidas previamente observadas da amostra;
6. Otimização de sistemas: minimização ou maximização de uma função objetivo obedecendo a restrições impostas ao problema;
7. Memórias associativas: recuperação de padrões corretos mesmo se os seus elementos constituintes forem apresentados de forma incerta ou imprecisa.

3.1. Rede Neural Perceptron Multicamadas (MLP)

Neste estudo, utilizou-se a rede perceptron multicamadas (MLP), que contém uma camada de entrada, uma oculta e uma camada de saída. Há apenas um neurônio na camada de saída, conforme mostrado na Figura 3.

Figura 3 - Configuração de neurônio utilizado na derivação do algoritmo *backpropagation*.



Fonte: Silva, Spatti e Flauzino (2010).

Após a entrada dos dados X_i tem-se o cálculo das ativações dos neurônios da camada escondida:

$$I_j^{(L)} = \sum_{i=0}^n W_{ji}^{(L)} X_i, \quad j = 1, \dots, n. \quad (11)$$

Sendo os $W_{ji}^{(L)}$ as matrizes de pesos cujos elementos denotam o valor do peso sináptico conectando o j -ésimo neurônio da camada (L) ao i -ésimo neurônio da camada (L-1), sendo n igual ao número de entradas e X_0 uma constante (*bias*).

Os $I_j^{(L)}$ são vetores cujos elementos denotam a entrada ponderada (combinação linear dos pesos e das entradas) em relação ao j -ésimo neurônio da camada L.

Os $Y_j^{(L)}$ são vetores cujos elementos representam a camada de saída do j-ésimo neurônio em relação à camada L, com o cálculo da função de ativação, tal que a função de ativação pode ser a função logística ou tangente sigmóide. A saída é definida por:

$$Y_j^{(L)} = g(I_j^{(L)}). \quad (12)$$

As camadas da rede MLP, neste trabalho, possuem funções de transferência tangente sigmóide na camada oculta e uma função de transferência linear na camada de saída.

A generalização de uma rede neural é definida como a capacidade de reconhecer novos conjuntos de dados. Para melhorar a generalização de uma rede geralmente os dados são divididos em treinamento, validação e conjuntos de testes.

O modelo MLP, implementado neste estudo, utilizou a função *newff*, do *software* MATLAB (MATHWORKS, 1995).

3.1.1. Rede perceptron multicamadas com o algoritmo de Levenberg-Marquardt

Segundo Silva, Spatti e Flauzino (2010) diversas variações do método *backpropagation* têm sido propostas com o objetivo de tornar o processo de convergência mais eficiente. Entre tais aperfeiçoamentos, tem-se o método de inserção do termo de momentum, o *resilient-propagation* e o Levenberg-Marquardt.

O algoritmo de Levenberg-Marquardt é um método gradiente de segunda ordem, baseado no método dos mínimos quadrados para modelos não lineares, que pode ser incorporado ao algoritmo *backpropagation* para potencializar a eficiência do processo de treinamento (SILVA; SPATTI; FLAUZINO, 2010).

Enquanto o algoritmo *backpropagation* é um método de descida no gradiente da função erro quadrático a fim de minimizá-la, o algoritmo de Levenberg-Marquardt é uma aproximação do método de Newton (DAN FORESEE; HAGAN, 1997).

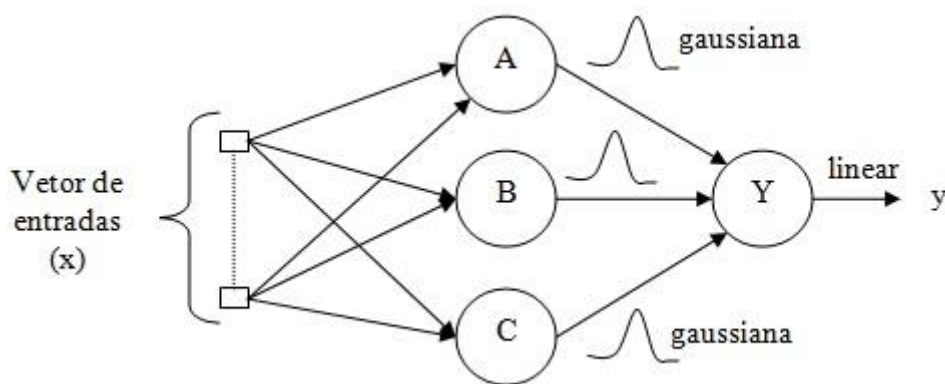
3.2. Rede Neural Função de Base Radial (RBF)

Rede neural de base radial (RBF) (RUSSEL; NORVIG, 2014) é uma rede *feedforward* multicamadas cujos neurônios possuem como função de ativação uma função de base radial. Geralmente utiliza uma função gaussiana (HAYKIN, 1999). É utilizada em aproximação de funções, predição de séries temporais e controle.

Duas das principais diferenças entre a RBF e a MLP é a função de transferência no interior da camada oculta (tangente sigmóide na MLP, gaussiana na RBF), e a camada de saída, Equação (18).

A arquitetura da rede RBF, conforme mostrado na Figura 4, tem como primeira camada vetores de entrada com dimensão m . A segunda camada é chamada de camada oculta (escondida), que é composta de uma função de base radial. Neste trabalho utilizou-se a gaussiana, na qual os neurônios realizam um mapeamento não linear.

Figura 4 - Arquitetura da RBF.



Fonte: Silva, Spatti e Flauzino (2010).

A camada de saída é meramente uma combinação linear das funções radiais de saída dos neurônios da camada escondida. Um único neurônio é utilizado, em seguida, para formar a camada de saída (combinação linear).

A rede neural RBF apresenta, principalmente, dois estágios. No primeiro, efetua-se a determinação dos parâmetros dos neurônios RBF, ou seja, os centros dos neurônios e a largura da função gaussiana, conforme mostrado na Equação (17). No segundo e último estágio, realiza-se o cálculo dos pesos entre a camada escondida e a camada de saída, de acordo com a Equação (18).

$$\varphi_i(x) = \exp\left(-\frac{\|x - u_i\|^2}{2\sigma_i^2}\right), \quad (17)$$

$$y = \sum_{i=1}^N w_i \varphi_i(x), \quad (18)$$

sendo que $x = [x_1, x_2, x_3, \dots, x_m]^T \in \mathbb{R}^m$ corresponde ao vetor de entrada, $u_i = [u_{i1}, u_{i2}, \dots, u_{im}]^T \in \mathbb{R}^m$ corresponde ao vetor dos centros e σ corresponde à largura do i -ésimo nó. O número de neurônios da camada escondida é indicado por N .

O modelo RBF implementado neste estudo utilizou a função *newrbe* disponível no *software* MATLAB (MATHWORKS, 1995).

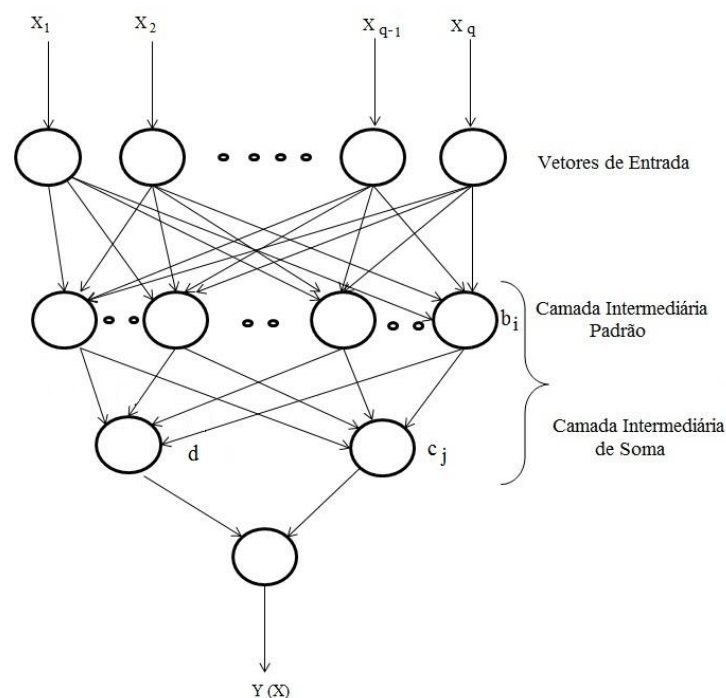
3.3 Rede Neural de Regressão Generalizada (GRNN)

A GRNN, proposta por Specht (1991), é uma RNA com estrutura do tipo não recorrente (*feedforward*) e requer treinamento supervisionado. Esta rede possui uma estrutura fixa, dada por uma camada de entrada, duas intermediárias (denominadas padrão e soma) e uma de saída, na qual o número de neurônios de cada camada depende do número de padrões de treinamento de entradas e saídas da rede. A função de ativação utilizada é a de base radial na camada de entrada padrão, como na rede RBF.

A principal vantagem da GRNN é a velocidade com que a rede pode ser treinada, e os valores obtidos no treinamento tornam-se vetores pesos nas camadas intermediárias.

Sua arquitetura pode ser vista na Figura 5.

Figura 5 - Arquitetura da rede neural GRNN.



Fonte: Specht (1991).

Na camada de entrada da GRNN tem-se n neurônios. A primeira camada intermediária é composta por n neurônios, chamada camada padrão, e a segunda camada intermediária, chamada de soma, é composta por s neurônios do tipo numerador e um neurônio do tipo denominador. A camada de saída é composta por s neurônios de saída.

As unidades de entrada fornecem as variáveis X para todos os neurônios da camada intermediária (padrão). A unidade padrão pode ser um exemplar ou um centro do cluster. Quando um novo vetor X entra na mesma rede ele é subtraído do vetor armazenado representando cada centro do cluster, Equação (19). Tanto o quadrado como os valores absolutos dessas diferenças são somados e alimentam uma função ativação não linear, normalmente a exponencial, na camada padrão, Equação (20). A saída desta camada é enviada para a camada de soma. As unidades de soma realizam o somatório do produto entre um vetor peso e um vetor obtido na saída da unidade padrão, Equação (21) e o somatório dos pesos Equação (22). Este número de observações varia de acordo com o número de saídas desejadas. As unidades de saída dividem cada resultado da camada de soma pelo somatório total de todas as unidades de soma, Equação (23) (SPECHELT, 1991).

$$a_i = \|X - X_i\|, \quad i = 1, \dots, n, \quad (19)$$

$$b_i = \exp\left(-\frac{a_i^2}{2\sigma^2}\right), \quad i = 1, \dots, n, \quad (20)$$

$$c_j = \sum_{i=1}^n y_{j,i} b_i, \quad j = 1, \dots, s, \quad (21)$$

$$d = \sum_{i=1}^n b_i, \quad i = 1 \dots n, \quad (22)$$

$$\hat{y}_j = \frac{c_j}{d}, \quad j = 1, \dots, s, \quad (23)$$

$$Y = \begin{bmatrix} \hat{y}_1, \dots, \hat{y}_s \end{bmatrix}^T. \quad (24)$$

A estrutura da GRNN empregada neste estudo utilizou a função *newgrnn* implementada no MATLAB (MATHWORKS, 1995).

3.4 Conclusões Parciais

Este capítulo apresentou conceitos sobre as redes neurais, suas arquiteturas e utilização.

Os estudos recentes na área da medicina propõem o uso das redes neurais para previsão de doenças.

Neste estudo apresentou-se as redes MLP, RBF e GRNN que podem ser utilizadas através de interfaces denominadas *toolboxes*, ou através da linguagem de *script* (programação) usando o *software* MATLAB (MATHWORKS, 1995). Neste trabalho utilizou-se a linguagem de *script* do próprio *software*, que possui comandos ou funções para cada rede neural. A validação das mesmas será realizada através de cálculos do RMSE (*Root Mean Square Error*) e da matriz de confusão utilizada para avaliar o desempenho de um classificador ou algoritmo supervisionado no reconhecimento de registros de diferentes classes.

4 DESENVOLVIMENTO DAS APLICAÇÕES UTILIZANDO ANFIS E REDES NEURAIIS

Umphred (2004) menciona que a hemiplegia é um sinal clássico de doença neurovascular do cérebro, sendo uma paralisia de um lado do corpo.

O Acidente Vascular Cerebral (AVC) é o maior causador de lesões no cérebro que originam uma hemiplegia. Sua causa é, normalmente, uma ruptura de um vaso (por aneurisma, hipertensão, etc.) ou por trombose de uma artéria (DAVIES, 1996).

No intuito de buscar soluções que melhorem a qualidade de vida de pacientes acometidos por algum nível de hemiplegia, esta aplicação pretende realizar uma análise comparativa entre várias redes neurais (MLP, GRNN e RBF) e rede ANFIS buscando identificar pacientes hemiplégicos ou não hemiplégicos.

O segundo exemplo de aplicação, na agronomia, proposto com o uso das redes neurais, tem a vantagem de prever ou estimar modelos não lineares, através de dados observados em cultivos anteriores. Mais especificamente, estimar a produção de trigo para diferentes doses de nitrogênio.

4.1 Métodos de Avaliação de Classificação dos Resultados

Han, Kamber e Pei (2006) dizem que a matriz de confusão analisa o desempenho de um classificador ou algoritmo, com aprendizado supervisionado, para reconhecer registros de diferentes classes. Uma matriz de confusão de m classes é uma tabela M , de tamanho m por m , na qual uma entrada M_{ij} indica o número de registros da classe i que foi rotulada pelo classificador como classe j . Para um classificador ter bom resultado, a maioria dos registros deve ser representada ao longo da diagonal principal da matriz de confusão, com o resto das entradas sendo igual a zero. A tabela pode ter outras linhas para apresentar o total (Tabela 1).

Tabela 1 - Exemplo de matriz de confusão em hemiplegia (diagnóstico).

Classificação	Não Hemiplegia	Hemiplegia
Não Hemiplegia	Verdadeiro Positivo (VP)	Falso Negativo (FN)
Hemiplegia	Falso Positivo (FP)	Verdadeiro Negativo (VN)

Fonte: Elaboração do próprio autor.

Nesta tabela:

- Verdadeiro positivo: refere-se ao registro que foi classificado corretamente pelo classificador;
- Verdadeiro negativo: refere-se ao registro negativo que foi classificado corretamente pelo classificador;
- Falso positivo: refere-se ao registro negativo que foi incorretamente classificado como positivo;
- Falso negativo: refere-se ao registro positivo que foi incorretamente classificado como negativo.

Para garantir que o resultado de um classificador proporcione uma estimativa confiável existem algumas técnicas de avaliação, dentre as quais a *Holdout* é muito utilizada (HAN; KAMBER; PEI, 2006).

No método *Holdout*, os dados fornecidos para classificação são aleatoriamente particionados em dois conjuntos independentes: conjunto de treinamento e conjunto de testes. Tipicamente, dois terços dos dados são alocados para o conjunto de treinamento e o restante é alocado para o conjunto de testes. O conjunto de treinamento é usado para gerar o modelo.

O desempenho dos classificadores foi avaliado utilizando-se: matriz de confusão, sensibilidade, especificidade e acurácia (KOHAVI; PROVOST, 1998). A raiz quadrada do erro médio quadrático (RMSE) também foi utilizada para representar o erro entre o critério desejado e a saída encontrada pela rede.

O desempenho individual de cada classificador, do modelo proposto, foi avaliado usando a sensibilidade (S_n) que está relacionada aos valores de verdadeiros positivos, a especificidade (S_p) que está relacionada aos valores de verdadeiros negativos, e acurácia (Acc) que está relacionada ao valor de acerto do classificador, e podem ser representadas como (KOHAVI; PROVOST, 1998):

$$S_n = \frac{VP}{VP + FN} , \quad (25)$$

$$S_p = \frac{VN}{VN + FP} , \quad (26)$$

$$Acc = \frac{VP + VN}{VP + VN + FN + FP} , \quad (27)$$

sendo que VP, FP, VN e FN representam o número de hígidos reconhecidos (não hemiplégicos), o número de hemiplégicos reconhecidos como hígidos, o número de

hemiplégicos reconhecidos corretamente, e FN representa o número de hígidos reconhecidos como hemiplégicos, respectivamente.

Neste estudo, a raiz quadrada do erro médio quadrático (RMSE) foi usada para calcular o erro das redes MLP, RBF, GRNN e ANFIS, de acordo com a Equação (28) (HAYKIN, 1998):

$$RMSE = \sqrt{\frac{1}{n} \sum_{i=1}^n (d_i - d'_i)^2} \quad (28)$$

na qual d_i é o vetor desejado, d'_i é o vetor previsto e n corresponde ao número de pares de entrada-saída do vetor usado no treinamento.

4.2 Desenvolvimento das Aplicações em Hemiplegia

O desenvolvimento das redes, nos estudos referentes à hemiplegia, foi dividido em duas fases: a fase de aprendizagem, em que o treinamento de rede adquiriu conhecimento sobre os dados vinculados a hemiplégicos ou não hemiplégicos, e a fase de classificação, na qual se identificou os pacientes como hemiplégicos ou não.

Para executar as etapas mencionadas, foram obtidas informações a partir de voluntários através de medições realizadas com 81 indivíduos normais e 10 hemiplégicos, sendo todos voluntários e acompanhados por profissionais de saúde, com a autorização do Comitê de Ética em Pesquisa da UNESP-Universidade Estadual Paulista, Presidente Prudente, São Paulo, Brasil, Processo nº 160/2007. O documento T.C.L.E. (Termo de Consentimento Livre e Esclarecido) foi informado a todos os voluntários.

Para a obtenção dos dados dos voluntários utilizou-se um baropodômetro, com um programa desenvolvido em LabVIEW e que continha 24 sensores para cada pé de paciente, implementado por Freitas (2008), que mediu a distribuição dos pesos na região plantar, com 81 pacientes hígidos (com nenhum histórico de dor ou problemas no sistema neuromuscular) e 10 hemiplégicos, com mais de um ano de hemiplegia, Figuras 18 e 19 (Apêndice A). Estes dados geraram um arquivo que contém 48 linhas diferentes para representar cada indivíduo (FREITAS, 2008).

Estes indivíduos foram separados em dois grupos. O grupo 1, utilizado na fase de treinamento, foi composto por 50 pessoas normais e 7 hemiplégicas. O grupo 2, utilizado na fase de validação, foi composto por 34 indivíduos, sendo 31 normais e 3 hemiplégicos. A classificação dos voluntários foi realizada por três profissionais da área de saúde.

Para efeito deste estudo comparativo apresentou-se neste trabalho uma metodologia para classificar pacientes em hígidos ou hemiplégicos, utilizando o sistema ANFIS (Adaptive *Neuro Fuzzy* Inference System), rede MLP com Levenberg-Marquardt, RBF e GRNN, sendo todos implementados no *software* MATLAB.

Para a estrutura das redes neurais utilizou-se:

1. Vetor de entrada: 48 variáveis de entrada correspondentes aos sensores do baropodômetro, havendo 91 registros subdivididos em conjunto de treinamento (57 registros) e validação (34 registros);
2. Vetor de saída: 1 variável correspondente ao valor que indica a classificação do voluntário. Os valores de saída são 0 e 1. Quando o valor estiver mais próximo de 0 indica indivíduo acometido com hemiplegia e quanto mais próximo de 1 indica indivíduo hígido;
3. 100 iterações para treinamento.

Do conjunto total de dados, 57 instâncias de dados que foram utilizados exclusivamente para treinamento e 34 instâncias de dados que foram usados exclusivamente para testes. Para evitar resultados de desempenho não confiáveis, em nenhum momento utilizou-se instâncias dos dados do conjunto de validação no procedimento de treinamento.

4.2.1 Estudo da hemiplegia com o ANFIS

O ANFIS (Adaptive *Neuro Fuzzy* Inference System) é um dos sistemas *neuro fuzzy* mais conhecidos e utilizados. A partir dos vetores de entradas e saída conhecidos, a ferramenta busca a elaboração de um modelo. Após a escolha do treinamento a ser utilizado pela rede, *backpropagation* ou híbrido, deve-se inserir o erro desejado e o número de épocas (iterações) realizadas para o treinamento. Outro item importante é determinar adequadamente o número de entradas e o número de funções de pertinência relacionadas com as mesmas.

A limitação desta técnica híbrida é a mesma para os sistemas nebulosos e não deve ser utilizada em sistemas com muitas entradas, em função do problema da explosão combinatória de regras. Supondo-se, por exemplo, uma rede *neuro fuzzy* com quatro variáveis de entrada, na qual cada uma delas tenha seu universo dividido em três conjuntos nebulosos ou funções de pertinência (mf), pode-se chegar a um total de 81 (3^4) regras.

Para utilizar o ANFIS nos dados de hemiplegia, utilizou-se nos dados de entrada a função *subtractive cluster*, presente no programa, que reduz o espaço de entradas em

agrupamentos, permitindo a implementação do sistema *neuro fuzzy* com um número elevado de entradas. Isto porque as variáveis de entrada, neste estudo, são 48, correspondentes aos 24 sensores que monitoram a descarga de peso em cada pé do paciente (direito e esquerdo).

A função de pertinência escolhida foi automaticamente a gaussiana após a seleção do tipo de partição do espaço de entrada via agrupamento (*subtractive cluster*). Para a função *subtractive cluster* especificou-se o raio de influência da função gaussiana (valor = 1,0), fator multiplicador (1,5), raio de aceitação no agrupamento (valor = 0,75) e raio de rejeição no agrupamento (0,25).

Os dados foram treinados com o algoritmo híbrido.

Vale ressaltar que no ANFIS, existem dois parâmetros para o critério de parada para o treinamento dos dados; o erro de tolerância (que deve ser um valor muito pequeno caso não se conheça o comportamento do erro de treinamento), e o número de iterações (que define o número de iteração máximo para o ANFIS convergir).

O erro de tolerância escolhido foi zero e o número de épocas desejadas para o treinamento foi 100, sendo este o valor utilizado como critério de parada para o modelo.

O Quadro 1 ilustra a especificação do sistema de inferência *fuzzy* no estudo sobre hemiplegia.

Quadro 1 - Especificação do sistema de inferência *fuzzy* no estudo sobre hemiplegia.

Rede Adaptativa	Topologia	Técnica de aprendizagem	Função MATLAB	Outras informações
ANFIS	<i>Feedforward</i> (Não recorrente)	Híbrido	<i>Anfisedit</i>	<ul style="list-style-type: none"> • 34 regras <i>fuzzys</i>; • Partição de entrada: função <i>Subtractive cluster</i>; • Parâmetros da função gaussiana: Raio de influência : 1,0; Fator multiplicador:1,5; Raio de aceitação: 0,75; Raio de rejeição: 0,25.

Fonte: Elaboração do próprio autor.

O sistema de inferência *neuro fuzzy* adaptativo foi implementado através da função *anfisedit* da toolbox *fuzzy logic* do MATLAB (MATHWORKS, 1995) e também pode ser utilizado através da linguagem de *script* do mesmo *software*. A Tabela 2 mostra os parâmetros do ANFIS utilizados no estudo da hemiplegia.

Tabela 2 - Parâmetro ANFIS para o estudo da hemiplegia.

Parâmetros ANFIS:	
Função de pertinência:	Gaussiana
Número de nós:	3383
Número de parâmetros lineares:	1666
Número de parâmetros não lineares:	3264
Iterações:	100 épocas
Número de pares de treinamento:	57 registros
Número de pares de validação:	34 registros
Saída na defuzificação:	Afim ($Z=Ax+By+C$)
Critério de Parada:	Erro=0
Total de regras <i>fuzzy</i> :	34
Algoritmo de Treinamento:	híbrido

Fonte: Elaboração do próprio autor.

4.2.2 Hemiplegia com redes neurais MLP, RBF e GRNN

Os estudos sobre a identificação de casos em hemiplegia também foram realizados com as redes neurais MLP, RBF e GRNN.

Hemiplegia e MLP

A topologia perceptron multicamadas *feedforward* foi adotada na construção da RNA devido à sua capacidade de representação de mapas funcionais não lineares entre as entradas e saídas. As redes neurais foram treinadas utilizando o treinamento Levenberg-Marquardt. A função tangente sigmóide foi utilizada para ativação dos neurônios na camada intermediária e função de ativação linear na camada de saída, Figura 19 (Apêndice A).

Utilizou-se 10 neurônios na camada intermediária (oculta).

Neste estudo, o modelo MLP foi implementado usando a função *newff* do *software* MATLAB (MATHWORKS, 1995).

Rede neural de base radial (RBF)

A topologia da rede neural RBF foi adotada na construção deste estudo comparativo para classificação de hemiplegia.

A função gaussiana foi utilizada para ativação dos neurônios na camada intermediária e função de ativação linear na camada de saída.

Utilizou-se 10 neurônios na camada intermediária (escondida), Figura 21 (Apêndice A).

Neste estudo, o modelo RBF foi implementado usando a função *newrbe* do *software* MATLAB (MATHWORKS, 1995).

Rede de regressão generalizada (GRNN)

A rede GRNN utiliza a função densidade probabilística, ou seja, usando a curva gaussiana, não necessita do gradiente descendente. Esse método é conhecido como aproximador universal de funções suaves, por isso é capaz de resolver qualquer problema de aproximação de função, uma vez que tenha dados suficientes (SPECHT, 1991).

A vantagem de se usar uma rede GRNN é sua velocidade de resposta, uma vez que outros métodos geralmente necessitam do gradiente descendente, o qual executa até a segunda derivada para achar os pontos de mínimos e máximos da função, gastando-se muito tempo para tal tarefa (HAYKIN, 1999).

Utilizou-se 10 neurônios na camada intermediária (escondida), Figura 20 (Apêndice A).

Neste estudo, o modelo GRNN foi implementado usando a função *newgrnn* do *software* MATLAB. O Quadro 2 mostra o resumo das estruturas das redes neurais utilizadas no estudo sobre hemiplegia.

Quadro 2 - Resumo das estruturas das redes neurais utilizadas no estudo sobre hemiplegia.

Rede Neural	Topologia da rede	Técnica de Aprendizagem	Número de neurônios camada escondida	Função MATLAB	Outras Informações
MLP	Multi camadas <i>feedforward</i>	Levenberg-Marquardt	10	<i>Newff</i>	Função de Ativação: <ul style="list-style-type: none"> • Tangente sigmóide (na camada oculta); • Função de Ativação: Linear (na camada de saída).
RBF	<i>Feedforward</i> (Não recorrente)	Gradiente descendente	10	<i>newrbe</i>	Função de Ativação: Função Gaussiana (na camada oculta); A saída é calculada com a combinação linear (na camada de saída).
GRNN	<i>Feedforward</i> (Não recorrente)	Gradiente descendente	10	<i>newgrnn</i>	Função de Ativação: Função Gaussiana (na camada oculta); A saída é calculada nas equações (22) e (23).

Fonte: Elaboração do próprio autor.

Foram realizados estudos das redes neurais com 20 neurônios na camada oculta e posteriormente com 10 neurônios na camada oculta. Observou-se que o desempenho das redes neurais com 10 neurônios na camada oculta foi superior e com erro RMSE menor.

4.2.3 Resultados sobre a classificação de hemiplegia

Nesta seção são apresentados os resultados da avaliação dos casos de pacientes hígidos e hemiplégicos nas redes *neuro fuzzy*, GRNN, RBF e MLP.

A Tabela 3 mostra o conjunto de dados de treinamento e validação.

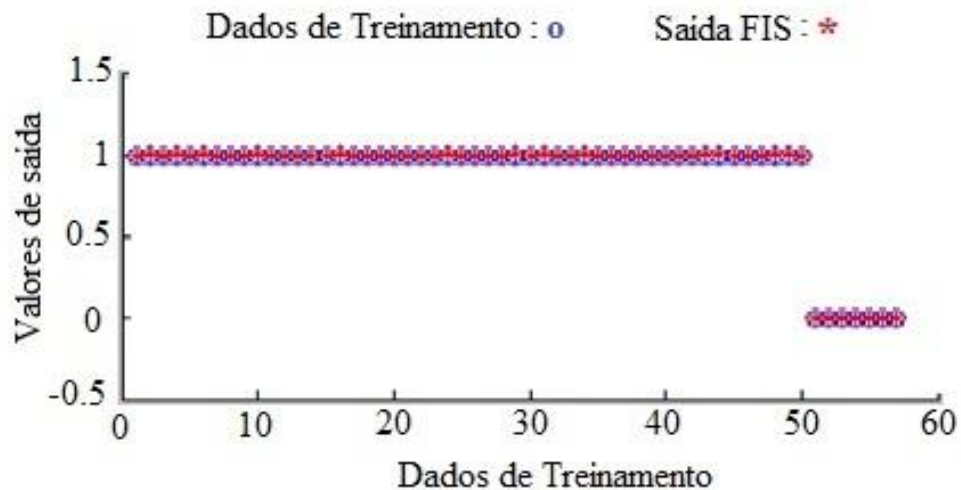
Tabela 3 - Conjunto de dados de treinamento e validação.

Conjunto de Dados	Não Hemiplégicos	Hemiplégicos	Total de indivíduos
Treinamento	50	7	57
Validação	31	3	34

Fonte: Elaboração do próprio autor.

A Figura 6 mostra o treinamento realizado no ANFIS. Este treinamento dos dados da hemiplegia foi realizado com êxito, pois o erro de treinamento com valor de $RMSE=0,00154$ evidencia que o sistema *neuro fuzzy* conseguiu aprender ou reconhecer com eficiência as amostras do treinamento.

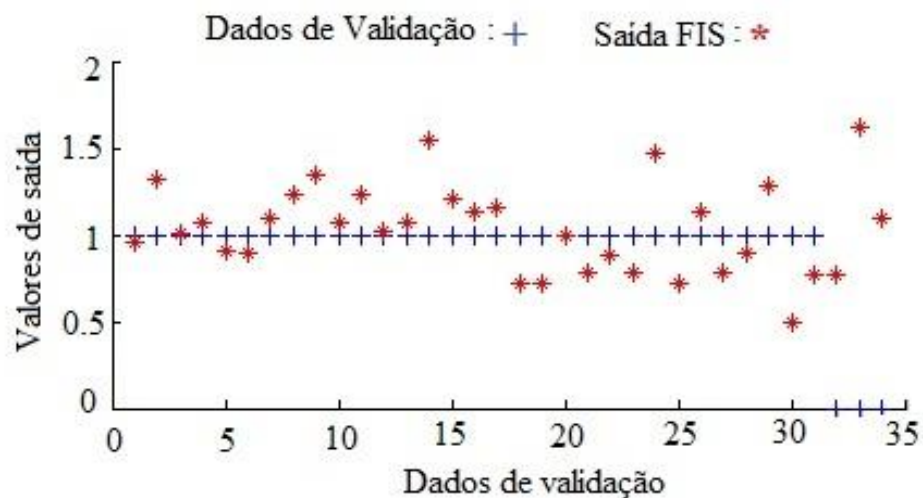
Figura 6 - Treinamento no ANFIS na classificação de hemiplegia.



Fonte: Elaboração do próprio autor.

A Figura 7 mostra o conjunto de validação, com 34 instâncias a serem validadas e a respectiva saída do ANFIS. A saída desejada era: (1) para hígidos e (0) para hemiplégicos. Observou-se que o SNF conseguiu identificar com eficiência os casos de hígidos, nas amostras de 1 a 31. Os casos de hemiplegia não foram identificados nas amostras de 32 a 34.

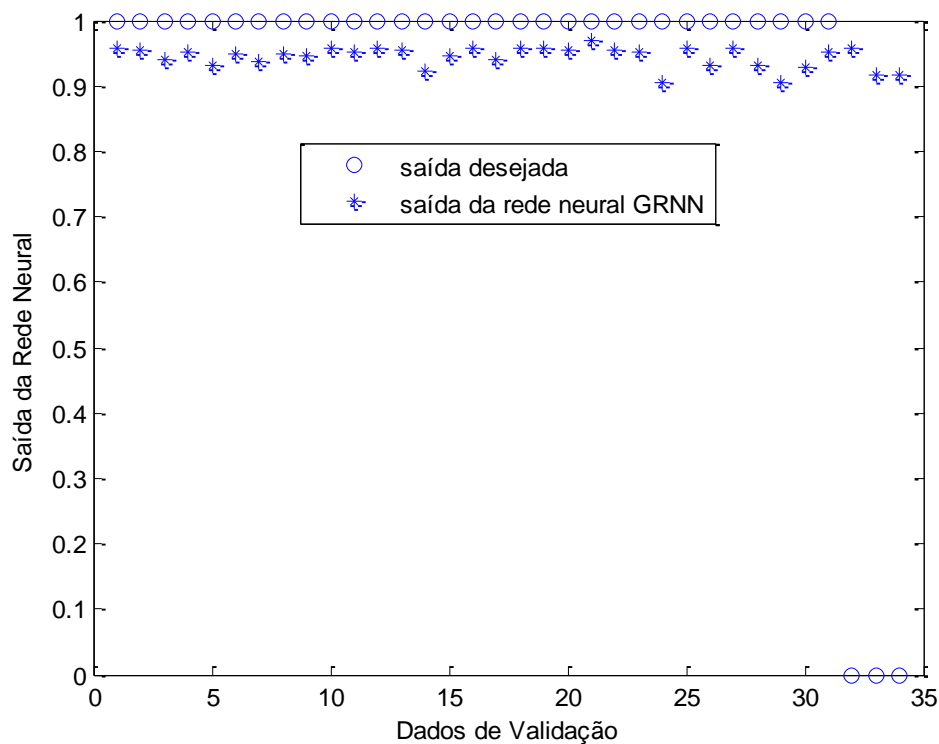
Figura 7 - Validação no ANFIS na classificação de hemiplegia.



Fonte: Elaboração do próprio autor.

Uma causa provável para este resultado é que o número de dados do arquivo de treinamento, usado para reconhecimento da rede, não tem número de classes balanceadas, ou seja, o mesmo número de parâmetros entre hígidos e hemiplégicos. Outra causa relevante para este fato é o número reduzido de pacientes hemiplégicos, no arquivo de treinamento. As classes desbalanceadas do arquivo de treinamento, com poucos dados de hemiplegia podem ter sido a causa desta situação. É importante notar que as amostras classificadas com valores de 0,5 para cima são considerados pacientes hígidos por esta rede. A Figura 8 apresenta a validação do conjunto de hemiplegia, com 34 instâncias pela rede neural GRNN. Observou-se que os casos de pacientes hígidos são identificados com eficiência porém os casos de hemiplegia não tiveram êxito.

Figura 8 - Validação da rede neural GRNN na classificação de hemiplegia.



Fonte: Elaboração do próprio autor.

Esta metodologia aprende com base nos exemplos que lhe são oferecidos no treinamento. Se a rede possui número reduzido de exemplos para reconhecimento, consequentemente não aprende e aumenta a probabilidade de erro. O que se pode verificar também é que os pacientes hígidos foram reconhecidos pela rede, fato este que favorece a afirmação anterior.

É importante salientar que a rede *neuro fuzzy* considerou como hígidos os pacientes com valores iguais ou acima de 0,5. Os pacientes considerados hemiplégicos ficaram com valores abaixo de 0,5.

A Figura 8 apresenta a validação dos dados de classificação de hemiplegia com a rede neural GRNN, evidenciando a mesma situação da rede anterior. Os pacientes hígidos foram identificados e os hemiplégicos não obtiveram êxito. Os prováveis motivos para esta situação são os mesmos descritos anteriormente para a rede *neuro fuzzy*.

O desempenho do ANFIS, avaliado através da matriz de confusão é mostrado nas Tabelas 4, 5, 6 e 7 (KOHAVI; PROVOST, 1998; HAN; KAMBER; PEI, 2006).

Tabela 4 - Matriz de Confusão do ANFIS na classificação de hemiplegia.

Classificação	Não Hemiplegia	Hemiplegia
Não Hemiplegia	(VP) 31	(FN) 0
Hemiplegia	(FP) 3	(VN) 0

Fonte: Elaboração do próprio autor.

Tabela 5 - Matriz de Confusão da rede MLP com treinamento Levenberg-Marquardt.

	Não Hemiplegia	Hemiplegia
Não Hemiplegia	(VP) 29	(FN) 2
Hemiplegia	(FP) 2	(VN) 1

Fonte: Elaboração do próprio autor.

Tabela 6 - Matriz de Confusão da rede RBF.

	Não Hemiplegia	Hemiplegia
Não Hemiplegia	(VP) 30	(FN) 1
Hemiplegia	(FP) 3	(VN) 0

Fonte: Elaboração do próprio autor.

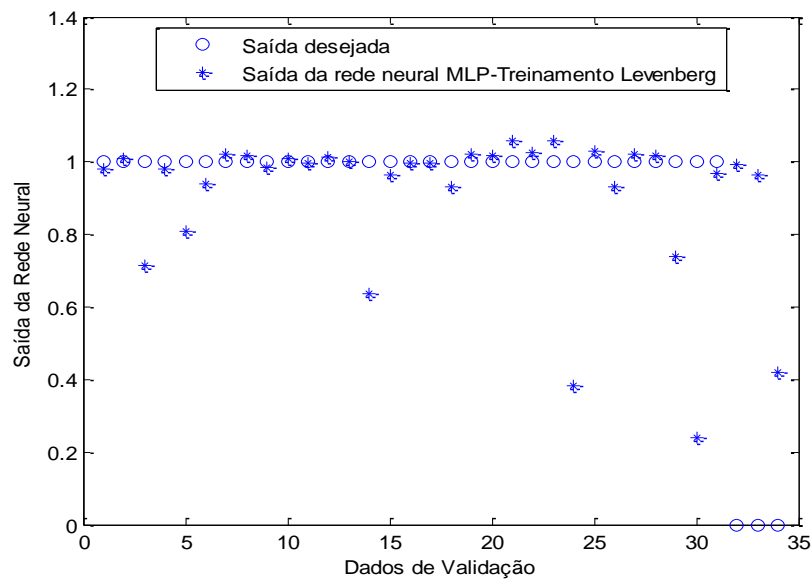
Tabela 7 - Matriz de Confusão da rede GRNN.

	Não Hemiplegia	Hemiplegia
Não Hemiplegia	(VP) 31	(FN) 0
Hemiplegia	(FP) 3	(VN) 0

Fonte: Elaboração do próprio autor.

A Figura 9 apresenta a validação do conjunto de classificação de hemiplegia, com 34 instâncias pela rede neural MLP. Observou-se que os casos de hígidos são identificados. A rede não reconheceu todos os casos de hemiplegia, mas acertou um indivíduo hemiplégico. Também errou caracterizando outros dois indivíduos hígidos como hemiplégicos, os falsos negativos (FN), nas amostras 24 e 30. Um ponto positivo neste erro foi a observação dos intervalos que a rede neural identifica como hemiplegia, os valores baixo de 0,5. Os valores iguais ou maiores que 0,5 são considerados indivíduos hígidos.

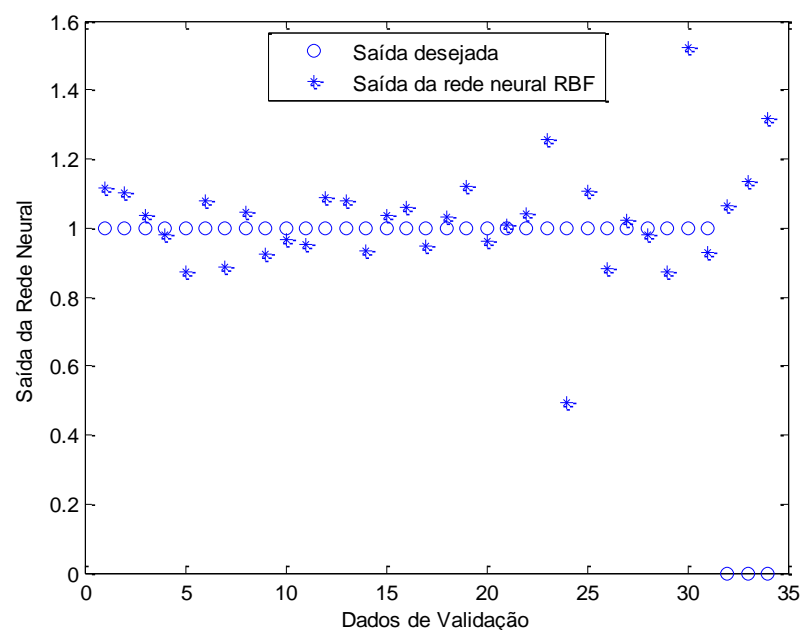
Figura 9 - Validação da rede neural MLP na classificação de hemiplegia.



Fonte: Elaboração do próprio autor.

A Figura 10 apresenta a validação do conjunto de hemiplegia, com 34 instâncias pela rede neural RBF. Observou-se que os casos de hígidos são identificados. A rede não reconheceu os casos de hemiplegia corretamente nos indivíduos que apresentavam o problema, porém revelou que o intervalo abaixo de 0,5 considera como hemiplegia ao identificar um falso negativo na amostra 24 (indivíduo hígido caracterizado como hemiplégico).

Figura 10 - Validação da rede neural RBF na classificação de hemiplegia.



Fonte: Elaboração do próprio autor.

A Tabela 8 mostra os cálculos de sensibilidade e especificidade das amostras de hemiplegia. O desempenho otimista observado na acurácia foi provocado pelas classes desbalanceadas. Sendo assim o classificador acertou quase o total da amostra de indivíduos hígidos e não acertou os casos de hemiplegia.

Tabela 8 - Performance de predição das redes neurais.

Redes Neurais	Sensitividade (Sn%)	Especificidade (Sp%)	Acurácia (Acc%)
MLP	93,5	33,3	88,2
RBF	96,8	0	88,2
GRNN	100	0	91,2
ANFIS	100	0	91,2

Fonte: Elaboração do próprio autor.

A Tabela 9 mostra a estrutura da rede neural e o resultado do erro RMSE.

Tabela 9 - Comparação do erro RMSE das redes neurais.

Redes Neurais	Número de entradas	Número de saídas	Número de épocas	RMSE
MLP	48	1	100	0,31
RBF	48	1	100	0,38
GRNN	48	1	100	0,28
ANFIS	48	1	100	0,43

Fonte: Elaboração do próprio autor.

Analisando os resultados obtidos, é possível constatar que as redes não classificaram corretamente os casos de hemiplegia. Conforme ampla discussão na literatura as redes neurais são metodologias que aprendem baseadas em experiências anteriores. Sendo assim, este fato provavelmente ocorreu devido ao reduzido número de amostras sobre os pacientes hemiplégicos. O número de amostras utilizadas para treinamento da rede neural foi 50 registros de pacientes hígidos e apenas 7 registros de pacientes hemiplégicos para treinamento das redes impossibilitando um aprendizado relevante para os indivíduos hemiplégicos. Mesmo assim a rede MLP conseguiu identificar um dos casos de hemiplegia corretamente. Um fato importante foi visualizar o intervalo em que rede neural considerou os pacientes como hígidos e como hemiplégicos.

Para validação do resultado e verificação da generalização da rede utilizou-se 34 registros, dos quais 31 eram pacientes hígidos e 3 hemiplégicos.

Freitas (2011) realizou um estudo com redes neurais MLP (com o treinamento *backpropagation*) e ARTMAP nebulosa na identificação de pacientes hemiplégicos e hígidos. Observou-se que a rede ARTMAP nebulosa, na fase de classificação, apresentou como normais todos os indivíduos hígidos (31) e dos três (3) hemiplégicos classificou corretamente 2 indivíduos. As duas redes neurais utilizadas apresentaram resultados satisfatórios. A rede neural ARTMAP nebulosa obteve as respostas com maior precisão do que a rede MLP, sendo que a primeira cometeu uma falha (reconheceu dois hemiplégicos) e a segunda MLP duas (reconheceu 1 hemiplégico) em 34 amostras.

A rede MLP com Levenberg-Marquardt proposta neste estudo obteve um desempenho equivalente à rede MLP com *backpropagation* de Freitas (2011).

O estudo indica que o uso das redes neurais para este tipo de aplicação é viável, desde que as classes de dados do arquivo de treinamento sejam balanceadas, ou seja, o mesmo número de indivíduos hígidos e hemiplégicos para que o classificador consiga aprender adequadamente. Um número expressivo de amostras também pode contribuir para um bom resultado.

4.3 Estudo comparativo de redes neurais para estimar a produção de trigo em função adubação nitrogenada

O trigo (*Triticum aestivum*) é uma planta cultivada durante o inverno e a primavera, nas regiões Sul (RS, SC e PR), Sudeste (MG e SP) e Centro-oeste (MS, GO e DF). Foi introduzido no Brasil pelos portugueses que costumam utilizar estes grãos na produção de pães, massas, bolos, biscoitos e ração para animais.

Este trabalho propõe uma nova metodologia, utilizando redes neurais artificiais, para estimar a produção média de grãos de trigo, utilizando experiências adquiridas anteriormente.

O desenvolvimento da aplicação abordará a implementação das redes neurais Perceptron Multicamadas (MLP), Rede de regressão generalizada (GRNN) e rede de base radial (RBF) na estimação da produção de trigo em função da adubação nitrogenada.

Para a implementação do treinamento das redes neurais utilizou-se dados experimentais do trabalho de Teixeira Filho et al. (2010), sobre uma cultura de trigo (*Triticum Aestivum*) registrada no período de dois anos na Fazenda Experimental da UNESP, em Selvíria-MS.

A validação desta metodologia foi realizada comparando-se os resultados das redes neurais propostas neste estudo com a rede *neuro fuzzy* apresentada no artigo de (SILVA et al., 2014) e curva de regressão quadrática proposta em (TEIXEIRA FILHO et al., 2010).

Em seus estudos sobre a cultura do trigo (TEIXEIRA FILHO et al., 2010) dizem que pequenas doses de nitrogênio limitam a produtividade e altas doses podem levar ao acamamento, dificultando assim a colheita, e provocando a queda de produtividade. Outro fator a considerar são que as altas doses do adubo podem causar prejuízos ao meio ambiente com a contaminação dos lençóis freáticos e os custos financeiros elevados para o produtor.

Para efeito de comparação das metodologias, os arquivos de treinamento das redes neurais foram elaborados com a mesma estrutura do trabalho de Silva et al. (2014), ou seja, com as doses de nitrogênio (0, 50, 100, 150 e 200 kg ha⁻¹) como variável de entrada e os valores médios de produção de trigo, referentes ao ano de 2006 e ano de 2007, como saída da rede neural, conforme a Tabela 10 e Tabela 11.

Tabela 10 - Dados de treinamento do ano de 2006, valores médios de produção de trigo por dose.

Doses de N (kg ha ⁻¹)	Produtividade de grãos (kg ha ⁻¹)
0	3029
50	3755
100	3791
150	3690
200	3543

Fonte: Silva et al. (2014).

Tabela 11 - Dados de treinamento do ano de 2007, valores médios de produção de trigo por dose.

Doses de N (kg ha ⁻¹)	Produtividade de grãos (kg ha ⁻¹)
0	2537
50	4164
100	4021
150	3858
200	3680

Fonte: Silva et al. (2014).

O arquivo de entrada das redes neurais utilizadas neste estudo com a linguagem de *script* do MATLAB, pode ser visto nas Figuras 22 a 27 no Apêndice A. Para cada rede neural foram realizados treinamentos dos anos de 2006 e 2007, contendo o vetor de entrada, no caso representado pela dose de nitrogênio, que corresponde a única variável de entrada e o vetor de saída desejado, referente à produção média de grãos de trigo.

A seguir, a Tabela 12 ilustra os detalhes sobre a estrutura das redes neurais para a aplicação na estimação da produtividade média dos grãos do trigo.

Tabela 12 - Estrutura das redes neurais.

Redes Neurais	Número de entradas	Número de saídas	Número de épocas
MLP	1	1	100
RBF	1	1	100
GRNN	1	1	100

Fonte: Elaboração do próprio autor.

O Quadro 3 mostra a arquitetura das redes neurais utilizadas para o modelo de estimação da produção média de grãos de trigo. Primeiramente realizou-se testes com 20 neurônios na camada oculta e posteriormente com 10 neurônios na camada oculta de todas as redes neurais. O resultado apontou que a topologia com 10 neurônios obteve RMSE (Raiz Quadrada do Erro Médio Quadrático) menor.

Quadro 3 - Arquitetura das redes neurais na produção média de trigo.

Rede Neural	Topologia da rede	Técnica de Aprendizagem	Número de neurônios camada escondida	Função MATLAB	Outras Informações
MLP	Multi camadas <i>feedforward</i>	Levenberg-Marquardt	10	<i>newff</i>	Função de Ativação: • Tangente sigmóide na camada oculta; • Função de transferência linear na camada de saída.
RBF	<i>Feedforward</i> (Não recorrente)	Gradiente descendente	10	<i>newrbe</i>	Função de Ativação: • Função Gaussiana (na camada oculta); • Combinação linear (na camada de saída).
GRNN	<i>Feedforward</i> (Não recorrente)	Gradiente descendente	10	<i>newgrnn</i>	Função de Ativação: • Função Gaussiana (na camada oculta); • A saída é calculada nas equações (22) e (23).

Fonte: Elaboração do próprio autor.

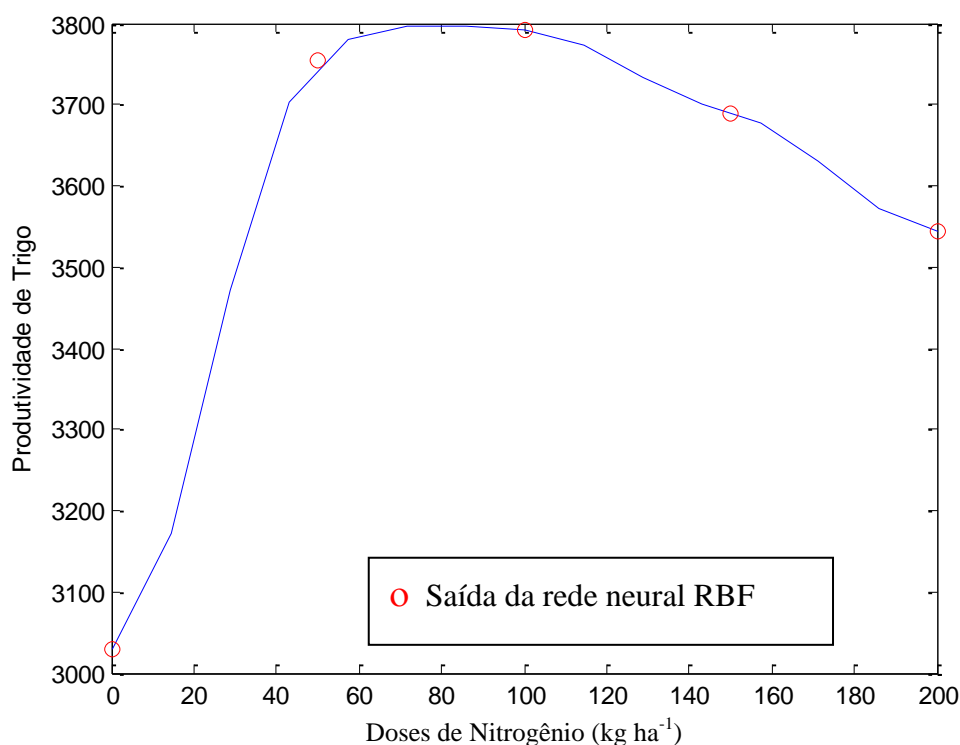
4.4 Resultados

Os resultados encontrados em relação à modelagem da estimativa da produção média de trigo com redes neurais podem ser analisados através dos gráficos a seguir.

Os gráficos evidenciam que esta nova metodologia de estimação obteve resultados melhores do que a metodologia de Silva et al (2014) com erro RMSE menores, com uma curva semelhante a uma gaussiana, típica para este tipo de estimação.

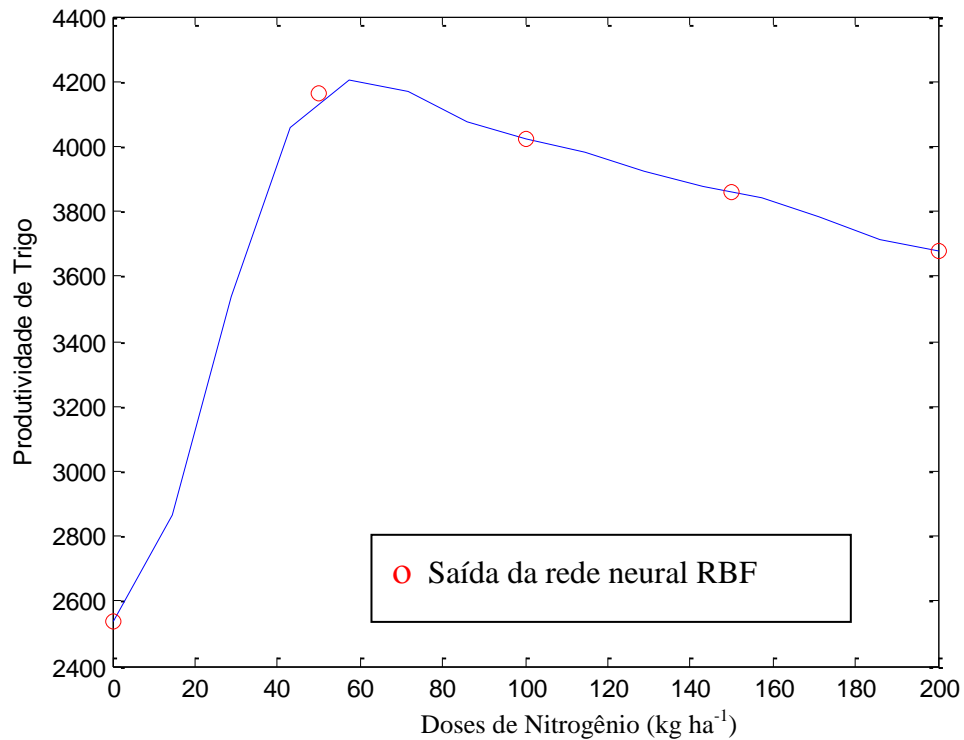
Nos gráficos das Figuras 11 e 12 ilustram o desempenho da rede neural RBF na estimativa de produção dos grãos de trigo, nos anos de 2006 e 2007, revelando o bom desempenho desta rede em função da curva real de produção. Os resultados da produtividade de trigo obtidos através das curvas utilizando a rede neural RBF dos anos de 2006 e 2007 apresentam os melhores resultados entre as doses de 60 a 100 (kg ha^{-1}) e 40 a 100 (kg ha^{-1}) respectivamente.

Figura 11 - Rede neural RBF na produção de trigo ano 2006.



Fonte: Elaboração do próprio autor.

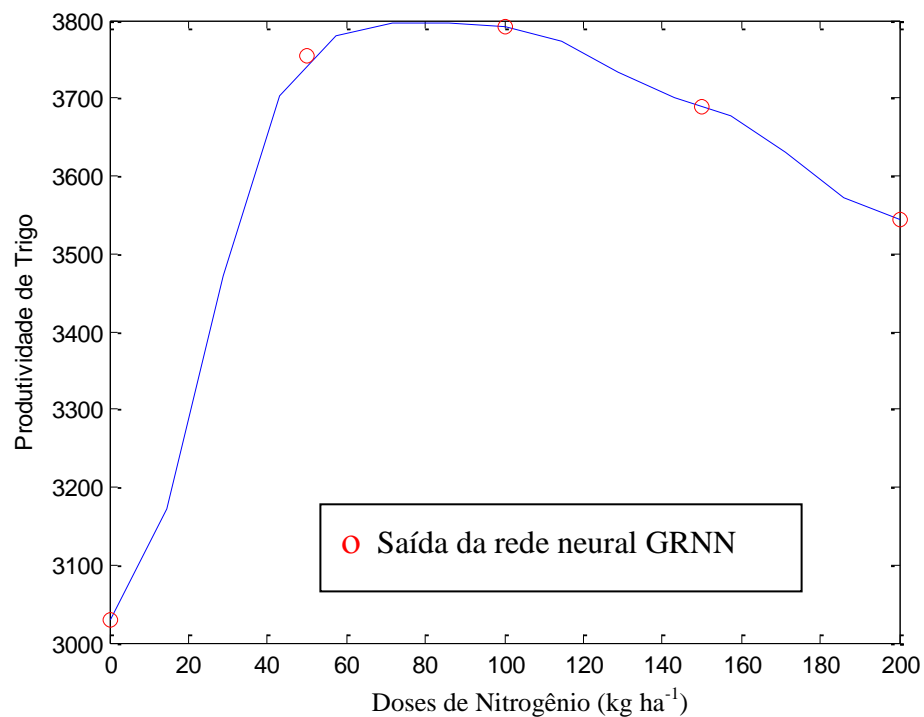
Figura 12 - Rede neural RBF na produção de trigo ano 2007.



Fonte: Elaboração do próprio autor.

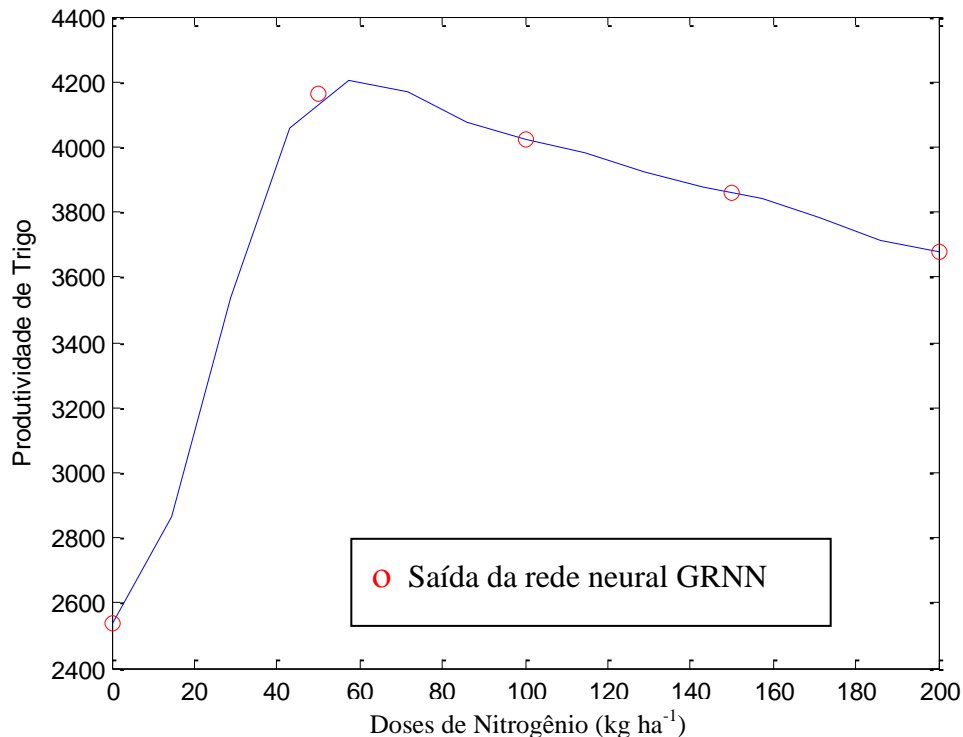
As Figuras 13 e 14 evidenciam o desempenho da rede neural GRNN na estimativa da produção de grãos de trigo com base nas doses de nitrogênio.

Figura 13 - Rede neural GRNN na produção de trigo ano 2006.



Fonte: Elaboração do próprio autor.

Figura 14 - Rede neural GRNN na produção de trigo ano 2007.



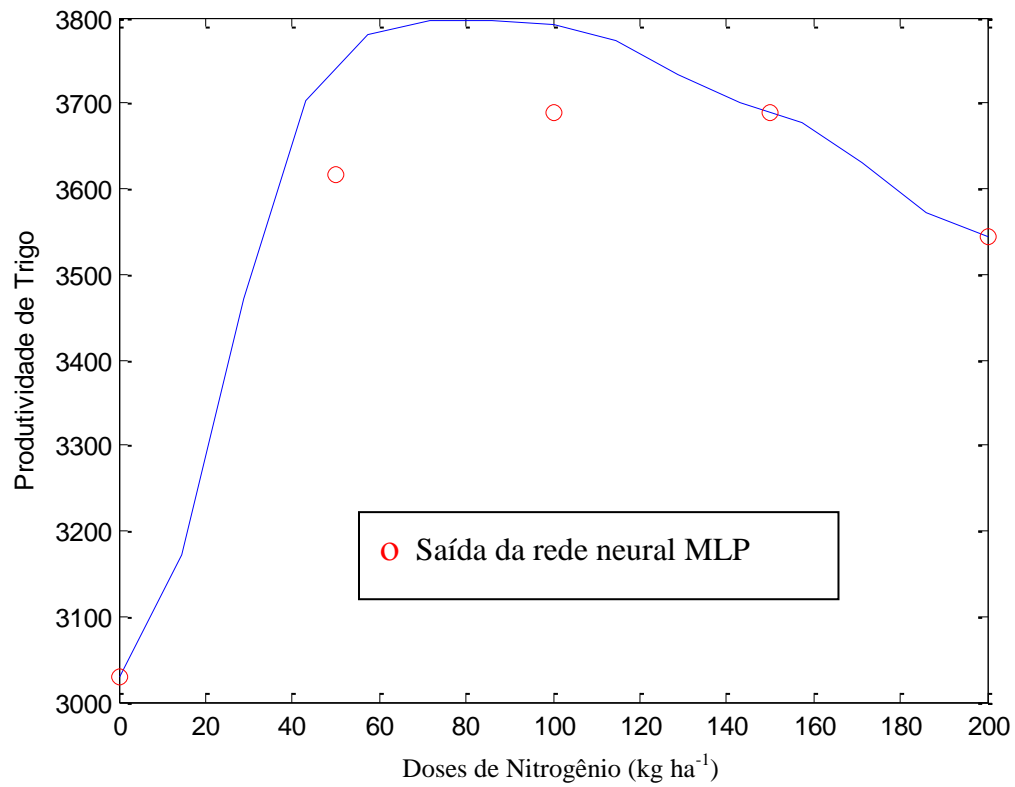
Fonte: Elaboração do próprio autor.

Os resultados da produtividade de trigo obtidos através das curvas utilizando a rede neural GRNN dos anos de 2006 e 2007 apresentam os melhores resultados entre as doses de 60 a 100 (kg ha⁻¹) e 40 a 100 (kg ha⁻¹) respectivamente.

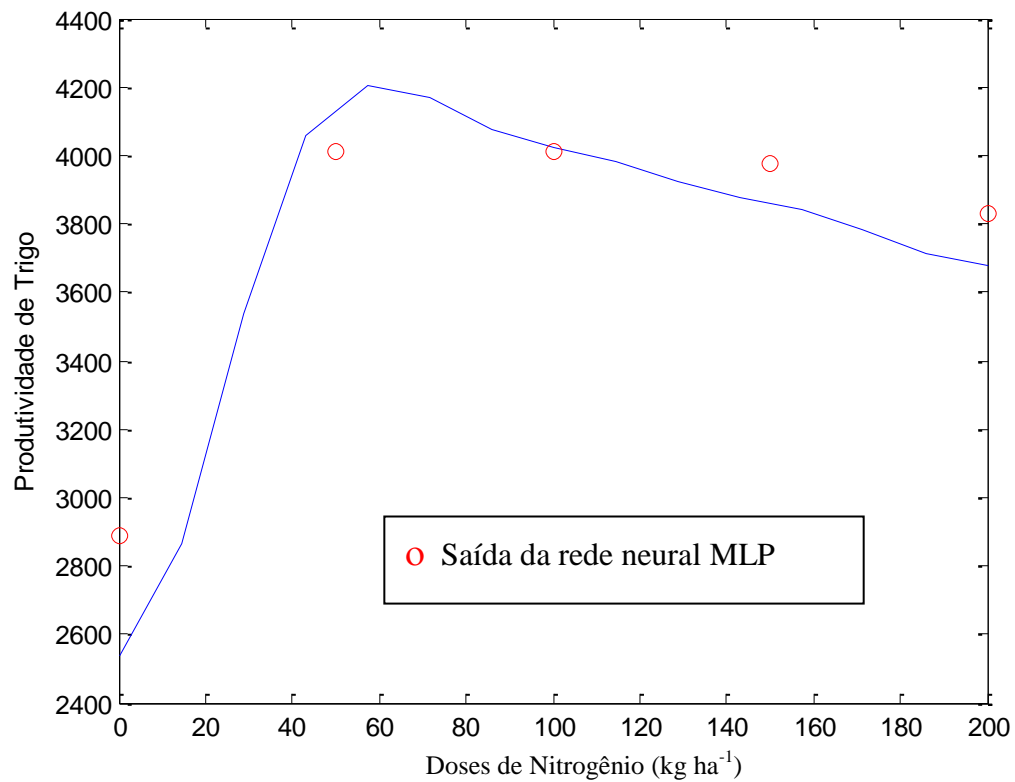
A rede GRNN apresentou bom nível de acerto em relação à curva real de produção e baixo erro RMSE.

As Figuras 15 e 16 mostram o desempenho da rede neural MLP. Dentre as redes estudadas nesta aplicação, a MLP foi a que apresentou o pior desempenho em relação a curva real de produção e também ao ser comparada com a rede *neuro fuzzy* do trabalho de Silva et al. (2014). As saídas de produção obtidas pela rede obtiveram erros RMSE mais altos que o erro da rede *neuro fuzzy* porém menores que a curva de regressão quadrática.

Os resultados da produtividade de trigo obtidos através das curvas utilizando a rede neural MLP com treinamento Levenberg-Marquardt, dos anos de 2006 e 2007 apresentam os melhores resultados entre as doses de 60 a 100 (kg ha⁻¹) e 40 a 100 (kg ha⁻¹) respectivamente.

Figura 15 - Rede neural MLP na produção de trigo ano 2006.

Fonte: Elaboração do próprio autor.

Figura 16 - Rede neural MLP na produção de trigo ano 2007.

Fonte: Elaboração do próprio autor.

As Tabelas 13 e 14, de Silva et al. (2014) mostram o valor do RMSE e os dados de produtividade de trigo estimados em função das doses de N (kg ha^{-1}), no modelo de regressão quadrática e no ANFIS.

Tabela 13 - Valor comparativo do RMSE nos modelos de regressão quadrática e ANFIS.

Ano da Produção	RMSE	
	Regressão Quadrática	ANFIS
2006	94,36	30,19
2007	265,14	71,76

Fonte: Silva et al. (2014).

Tabela 14 - Valores de produção no ANFIS, regressão quadrática e valores médios de produção real por dose.

Modelo gerado	Produtividade de grãos de trigo em função de N (kg ha^{-1})					
	Doses de N	0	50	100	150	200
Valor médio de produção real por dose	Safra 2006	3029	3755	3791	3690	3543
	Safra 2007	2537	4164	4021	3858	3680
ANFIS	Safra 2006	2980	3800	3796	3693	3533
	Safra 2007	2427	4280	4014	3860	3668
Regressão quadrática	Safra 2006	3100,76	3600,19	3830,63	3792,06	3484,5
	Safra 2007	2737,87	3713,84	4171,31	4110,28	3530,75

Fonte: SILVA et al. (2014).

Na Tabela 15 verifica-se o erro RMSE calculado com as redes neurais MLP, GRNN e RBF durante a estimativa da produção de trigo. Com esta tabela verifica-se que o desempenho das redes neurais GRNN e RBF foram superiores a metodologia de Silva et al. (2014), no qual compara-se o ANFIS com regressão quadrática. A rede neural MLP obteve o desempenho inferior ao ANFIS porém superior a curva de regressão quadrática.

Tabela 15 - Valor comparativo do RMSE nos modelos de regressão quadrática e ANFIS.

Ano da Produção	RMSE		
	GRNN	RBF	MLP
2006	0,0091	0,000016	62,94
2007	0,0216	0	192,03

Fonte: Elaborado pelo próprio autor

A Tabela 16 mostra os dados de saída estimados de produtividade de trigo com as redes neurais: GRNN, RBF e MLP.

Tabela 16 - Estimativa de produção média de grãos de trigo com redes neurais.

Modelo gerado	Produtividade de grãos de trigo em função de N (kg ha ⁻¹)					
	Doses de N	0	50	100	150	200
GRNN	Safra 2006	3029	3755	3791	3690	3543
	Safra 2007	2537	4164	4021	3858	3680
RBF	Safra 2006	3029	3755	3791	3690	3543
	Safra 2007	2537	4164	4021	3858	3680
MLP	Safra 2006	3026	3661	3685	3684	3548
	Safra 2007	2889	4011	4011	3971	3830

Fonte: Elaborado pelo próprio autor

4.5 Conclusões parciais

O exemplo de aplicação sobre hemiplegia mostrou a utilização do ANFIS com a partição de entrada *Subtractive Cluster*. O ANFIS tem uma limitação com relação ao número de entradas do sistema. Para uma aplicação com até nove entradas utiliza-se a partição *grid* (*grid partition*). Acima de nove entradas, deve-se utilizar o agrupamento subtrativo (*subtractive cluster*), desta forma evita-se o excesso de regras *fuzzy* geradas. No caso da hemiplegia utilizou-se esta opção uma vez que o modelo da aplicação possui 48 entradas e uma saída.

Com estes exemplos conclui-se que uma das vantagens do uso do sistema *neuro fuzzy* é o mapeamento de modelos não lineares quando não se tem especialistas para desenvolver as regras do modelo *fuzzy*, nem informações a respeito do sistema a ser modelado, mas tem-se um conjunto de treinamento sobre o mesmo, ou seja, pares de entrada e saída.

As definições dos parâmetros iniciais do ANFIS são simples e facilitam o desenvolvimento dos arquivos de treinamento e da validação.

Dentre as desvantagens no uso do ANFIS estão a definição das variáveis de entrada e seus conjuntos *fuzzy*, pois há dependência de um especialista ou de conhecimento sobre a aplicação. As funções de pertinência também precisam ser testadas a fim de encontrar a que melhor se ajusta aos parâmetros de forma a minimizar o erro na saída do ANFIS para identificar o modelo.

No exemplo de aplicação do ANFIS sobre hemiplegia, o objetivo era identificar pacientes hemiplégicos e não hemiplégicos. O objetivo foi parcialmente atingido devido a

poucas amostras de indivíduos hemiplégicos para treinamento das redes, não foi possível obter bons resultados na identificação dos mesmos. Observou-se que para o caso de hígidos no qual o número de amostras era maior a identificação foi satisfatória.

Com o uso das redes neurais em estudos sobre hemiplegia observou-se que a limitação dos dados de pacientes hemiplégicos prejudicou o desempenho das redes. Fato este observado nas matrizes de confusão das redes MLP, RBF e GRNN, em que os casos de hígidos obtiveram êxito e os casos de hemiplegia não obtiveram. Apenas uma das redes neurais, a MLP, conseguiu identificar um dos casos.

Na aplicação sobre agronomia buscou-se estimar a produção de trigo utilizando um estudo comparativo das redes neurais MLP, GRNN e RBF. Os resultados foram satisfatórios e mostraram RMSE menores do que o estudo de Silva et al. (2014), nas redes RBF e GRNN, quando esse modelo foi comparado com ANFIS e com a regressão quadrática, comumente utilizada para este fim.

5 CONSIDERAÇÕES FINAIS

Neste capítulo são apresentadas as considerações finais para as metodologias propostas neste trabalho, bem como trabalhos futuros e publicações.

5.1 Conclusões

Esta tese apresentou aplicações para a área de engenharia biomédica e agronomia utilizando para isto o sistema de inferência *neuro fuzzy* (ANFIS) e redes neurais artificiais.

Os resultados mostraram que o treinamento MLP com treinamento Levenberg-Marquardt teve a capacidade de generalização com RMSE de 0,31, enquanto que para a rede GRNN, o RMSE estava perto de 0,28. A rede RBF apresentou o resultado do RMSE equivalente a 0,38 e o ANFIS com RMSE 0,43.

A precisão para a classificação de hemiplegia, na matriz de confusão, atingiu 91,2% na GRNN e ANFIS, 88,2% para RBF e 85,3% para MLP com Levenberg-Marquardt, indicando que o método proposto é uma ferramenta viável para a identificação de hemiplegia. Apesar de se obter bons resultados de sensibilidade, também se obteve resultados não satisfatórios de especificidade. Este resultado provavelmente ocorreu porque existiam poucos dados sobre indivíduos hemiplégicos. Uma possível solução para a obtenção de melhores resultados em pesquisas futuras seria coletar mais dados durante a utilização do baropodômetro, sobretudo de indivíduos hemiplégicos, para os dados de treinamento e aprendizagem da rede neural.

O modelo de estimação de produção de trigo foi elaborado utilizando redes neurais e mostrou-se superior aos estudos realizados por Silva et al. (2014) onde utilizou-se ANFIS e curva de regressão quadrática. Os erros RMSE das redes RBF (0,000016 e 0), GRNN (0,0091 e 0,0216) foram muito menores que os erros encontrados na estimativa de produção com ANFIS e regressão quadrática. A rede MLP mostrou desempenho inferior ao ANFIS com RMSE (62,94 e 192,03) maiores que o mesmo, porém desempenho superior a curva de regressão quadrática com erro menor que a mesma.

Com esta aplicação verificou-se que as redes neurais podem ser uma metodologia viável para estimar a produção média de grãos de trigo e contribuir na área de agronomia.

5.2 Trabalhos futuros

Como sugestões para trabalhos futuros em hemiplegia têm-se:

1. Considerar outros requisitos diferentes dos pacientes, por exemplo, o lado onde ocorreu a hemiplegia, o fator de escala de Berg, a razão da hemiplegia, e o tempo decorrido de tratamento, analisar risco de queda para idosos, entre outros.
2. Espera-se que o modelo apresentado possa fornecer novas perspectivas sobre a investigação em hemiplegia.

Para agronomia sugere-se como trabalhos futuros:

1. Utilizar outras variáveis, ou parâmetros relevantes para a produção, como tipo de solo, cultivares e clima para avaliar a produtividade de uma cultura.
2. Estimar a produção de outras culturas com outras redes neurais, e máquina vetor de suporte (SVM).

5.3 Publicações

SILVA, A. A. V.; SILVA, I. A. F.; TEIXEIRA FILHO, M. C. M.; BUZETTI, S.; TEIXEIRA, M. C. M. Estimativa da produtividade de trigo em função da adubação nitrogenada utilizando modelagem neuro fuzzy. **Revista Brasileira de Engenharia Agrícola e Ambiental**, Campina Grande, v. 18, n. 2, p. 180–187, 2014 (coautoria).

SILVA, A. A. V.; SILVA, I. A. F.; TEIXEIRA, M. C. M.; SILVA, M. M. V.; MARQUES, J. C. C. Sistema fuzzy para diagnóstico de hérnias da região inguinal. In: CONGRESSO BRASILEIRO DE SISTEMA FUZZY-CBSF, 2., 2012, Natal. **Anais...** Natal: SBMAC, 2012. p. 919-931 (coautoria).

SILVA, I. A. F.; SILVA, A. A. V.; TEIXEIRA, M. C. M. Modelagem fuzzy com técnicas de agrupamento fuzzy e estruturas neuro fuzzy. In: CONGRESSO BRASILEIRO DE SISTEMA FUZZY-CBSF, 2., 2012, Natal. **Anais...** Natal: SBMAC, 2012. p. 968- 979 (autor principal).

REFERÊNCIAS

- AL-GAADI, K. A.; ABOUKARIMA, A. M.; SAYEDAHMED, A. A. Employing an adaptive neuro-fuzzy inference system for optimum distribution of liquid pesticides. **African Journal of Agricultural Research**, Victoria Island, v. 6, n. 9, p. 2078-2085, 2011.
- ALVAREZ, R. Predicting average regional yield and production of wheat in the Argentine Pampas by an artificial neural network approach. **European Journal of Agronomy**, Amsterdam, v. 30, n. 2, p. 70 -77, 2009.
- BABUSKA, R.; VERBRUGGEN, H. Neuro fuzzy methods for nonlinear system identification. **Annual reviews in control**, Kidlington, v. 27, n. 1, p. 73-85, 2003.
- BENINI, L. C. **Estimação da densidade de solos utilizando sistemas de inferência fuzzy**. 2007. 210 f. Tese (Doutorado em Agronomia/ Energia na Agricultura) – Faculdade de Ciências Agrônômicas, Universidade Estadual Paulista, Botucatu, 2007.
- BONISSONE, P. P. **Soft computing: the convergence of emerging reasoning technologies: soft computing**. Berlin: Springer Berlin Heidelberg, 1997. v. 1, p. 6-18.
- BUCENE, L. C.; RODRIGUES, L. H. A. Utilização de redes neurais artificiais para a avaliação de produtividade do solo, visando classificação de terras para irrigação. **Revista Brasileira de Engenharia Agrícola e Ambiental**, Campina Grande, v. 8, n. 2-3, p. 326-329, 2004.
- CARVALHO, C. C. N.; ROCHA, W. F.; UCHA, J. M. Mapa digital de solos: uma proposta metodológica usando inferência fuzzy. **Revista Brasileira de Engenharia Agrícola e Ambiental**, Campina Grande, v. 13, n. 1, p. 46-55, 2009.
- CHEN, J.; XI, Y.; ZHANG, Z. A clustering algorithm for fuzzy model identification. **Fuzzy Sets and Systems**, New York, v. 98, n. 3, p. 319-329, 1998.
- CHEN, Y. et al. Neural network and fuzzy control in FES-assisted locomotion for the hemiplegic. **Journal of Medical Engineering & Technology**, London, v. 28, n. 1, p. 32-38, 2004.
- CHIU, S. L. A cluster estimation method with extension to fuzzy model identification. In: CONFERENCE ON FUZZY SYSTEMS/IEEE WORLD CONGRESS ON COMPUTATIONAL INTELLIGENCE, 3., 1994, Orlando. **Proceedings...** New York: IEEE, 1994. p. 1240-1245.
- CHOI, J.; OH, S.; PEDRYCZ, W. Identification of fuzzy models using a successive tuning method with a variant identification ratio. **Fuzzy Sets and Systems**, New York, v. 159, n. 21, p. 2873-2889, 2008.
- DAN FORESEE, F.; HAGAN, M. T. Gauss-newton approximation to bayesian learning. In: IEEE INTERNATIONAL CONFERENCE ON NEURAL NETWORKS, 1997, 1997, Houston. **Proceedings...** New York: IEEE, 1997. p. 1930-1935.

DAS, R.; TURKOGLU, I.; SENGUR, A. Effective diagnosis of heart disease through neural networks ensembles. **Expert Systems with Applications**, New York, v. 36, n. 4, p. 7675–7680, 2009.

DAVIES, P. M. **Exatamente no centro: atividade seletiva do tronco no tratamento da hemiplegia no adulto**. São Paulo: Manole, 1996. 284 p.

DOMÍNGUEZ HERNÁNDEZ, K. R. et al. Development of an expert system as a diagnostic support of cervical cancer in atypical glandular cells, based on fuzzy logics and image interpretation. **Computational and Mathematical Methods in Medicine**, New York, v. 2013, n. 796387, p.1-17, 2013.

FREITAS, L. P. S. **Discriminação entre pacientes normais e hemiplégicos utilizando plataforma de força e redes neurais**. 2011. 83 f. Dissertação (Mestrado em Engenharia Elétrica)-Faculdade de Engenharia, Universidade Estadual Paulista, Ilha Solteira, 2011.

FREITAS, R. L. B. **Plataforma de força para aplicações biomédicas**. 2008. 136 f. Dissertação (Mestrado em Engenharia Elétrica)- Faculdade de Engenharia, Universidade Estadual Paulista, Ilha Solteira, 2008.

GAINO, R. **Controle de movimentos de pacientes paraplégicos utilizando modelos fuzzy T-S**. 2009. 182 f. Tese (Doutorado em Engenharia Elétrica)- Faculdade de Engenharia, Universidade Estadual Paulista, Ilha Solteira, 2009.

GAINO, R.; TEIXEIRA, M. C. M.; CARVALHO, A. A. DE; ASSUNÇÃO, E.; CARDIM, R.; SANCHES, M. A. A.; COVACIC, M. R. Realimentação derivativa e modelo fuzzy takagi-sugeno para controle da articulação do joelho de pacientes paraplégicos com o uso de acelerômetros. **Revista Brasileira de Engenharia Biomédica**, Rio de Janeiro, v. 27, n. 2, p. 67-78, 2011.

GOMEZ-SKARMETA, A. F.; DELGADO, M.; VILA, M. A. About the use of fuzzy clustering techniques for fuzzy model identification. **Fuzzy Sets and Systems**, New York, v. 106, n. 2, p. 179-188, 1999.

GULER, I.; UBEYLI, E. D. Detection of ophthalmic arterial doppler signals with Behcet disease using multilayer perceptron neural network. **Computers in Biology and Medicine**, Oxford, v. 35, n. 2, p. 121-132, 2005.

HAN, J.; KAMBER, M.; PEI, J. **Data mining: concepts and techniques**. 2 nd ed. San Francisco: Elsevier, 2006. 800 p.

HAYKIN, S. **Neural networks: a comprehensive foundation**. 2nd ed. New Jersey: Prentice Hall, 1998. 842 p.

HOSHI, K.; KAWAKAMI, J.; KUMAGAI, M.; KASAHARA, S.; NISIMURA, N.; NAKAMURA, H. An analysis of thyroid function diagnosis using Bayesian-type and SOM-type neural networks. **Chemical and Pharmaceutical Bulletin**, Tokyo, v. 53, n. 12, p. 1570 - 1574, 2005.

INSTITUTO BRASILEIRO DE GEOGRAFIA E ESTATÍSTICA – IBGE. **Projeção da população do Brasil por sexo e idade para o período 2000/ 2060**. [S. l.: s. n.], 2013. Disponível em:

<http://www.ibge.gov.br/home/estatistica/populacao/projecao_da_populacao/2013/default.shtm>. Acesso em: 22 mar. 2014.

JAIN, A. K.; MURTY, M. N.; AND FLYNN, P. J. Data clustering: a review. **ACM Comput. Surv.**, New York, v. 31, n. 3, p. 264–323, 1999.

JANG, J. R. ANFIS: adaptive-network-based fuzzy inference system. **IEEE Transactions on Systems, Man and Cybernetics**, New York, v. 23, n. 3, p. 665-685, 1993.

JANG, J.-SR; SUN, CHUEN-TSAI. Neuro-fuzzy modeling and control. **Proceedings of the IEEE**, New York, v. 83, n. 3, p. 378-406, 1995.

KASHANINEJAD, M.; DEHGHANI, A.A.; KASHIRI, M. Modeling of wheat soaking using two artificial neural networks (MLP and RBF). **Journal of Food Engineering**, Oxford, v. 91, n. 4, p. 602-607, 2009.

KOHAVI, R.; PROVOST, F. **On applied research in machine learning. Editorial for the special issue on applications of machine learning and the knowledge discovery process.** New York: Columbia University, 1998. v. 30, p. 127-132.

LAU, H.; TONG, K.; ZHU, H. Support vector machine for classification of walking conditions of persons after stroke with dropped foot. **Human movement science**, Amsterdam, v. 28, n. 4, p. 504-514, 2009.

LESSA, I. Epidemiologia das doenças cerebrovasculares no Brasil. **Rev Soc Cardiol Estado São Paulo**, São Paulo, v. 9, n. 4, p. 509-511, 1999.

MATHWORKS. **The student edition of MATLAB**: version 4 user's guide. Englewood Cliffs: Prentice Hall, 1995. 834 p.

OHNO-MACHADO, L.; MARK, A. Sequential versus standard neural networks for pattern recognition: an example using the domain of coronary heart disease. **Computers in Biology and Medicine**, Oxford, v. 27, n. 4, p. 267-28, 1997.

PAN, S. et al. Parkinson's disease tremor classification – a comparison between support vector machines and neural networks. **Expert Systems with Applications**, New York, v. 39, n. 12, p. 10764-10771, 2012.

PAPADAKIS, S. E.; THEOCHARIS, J. B. A GA-based fuzzy modeling approach for generating TSK models. **Fuzzy Sets and Systems**, New York, v. 131, n. 2, p. 121-152, 2002.

PETROSIAN, A. A.; PROKHOROV, D. V.; LAJARA-NANSON, W.; SCHIFFER, R. B. Recurrent neural network-based approach for early recognition of Alzheimer's disease in EEG. **Clinical Neurophysiology**, Shannon, v. 112, n. 8, p. 1378-1387, 2001.

PRIYONO, A.; ALIAS A. J.; RAHMAT, R. A. O.; HASSAN, A.; ALI, M. A. M. Generation of fuzzy rules with subtractive clustering. **Journal Teknologi**, Malaysia, v. 43, n. 1, p. 143-153, 2005.

RAJESWARI, K.; VAITHIYANATHAN, V.; NEELAKANTAN, T. R. Feature selection in ischemic heart disease identification using feed forward neural networks. **Procedia Engineering**, Amsterdam, v. 41, , p. 1818 – 1823, 2012.

- RUANO, M. G.; RUANO, A. E. On the use of artificial neural networks for biomedical applications. In: INTERNATIONAL WORKSHOP SOFT COMPUTING APPLICATIONS, 5., 2012, Szeged. **Proceedings...** Berlin: SPRINGER-VERLAG BERLIN, 2013. p. 433-451.
- RUSSEL, S. E.; NORVIG, P. **Artificial Intelligence: a modern approach**. 3rd ed. Harlow: Pearson, 2014. 1079 p.
- SHAW, S.; SIMÕES, M. G. **Controle e modelagem fuzzy**. São Paulo: Edgar Blucher, 1999. 166 p.
- SHEPPARD, D.; MCPHEE, D.; DARKE, C.; SHRETHRA, B.; MOORE, R.; JUREWITZ, A.; GRAY, A. Predicting cytomegalovirus disease after renal transplantation: an artificial neural network approach. **International Journal of Medical Informatics**, Clare, v. 54, n. 1, p. 55–76, 1999.
- SILVA, A. A. V.; SILVA, I. A. F.; TEIXEIRA, M. C. M.; SILVA, M. M. V.; MARQUES, J. C. C. Sistema fuzzy para diagnóstico de hérnias da região inguinal. In: CONGRESSO BRASILEIRO DE SISTEMA FUZZY-CBSF, 2., 2012, Natal. **Anais...** Natal: SBMAC, 2012. p. 919-931.
- SILVA, A. A. V.; SILVA, I. A. F.; TEIXEIRA FILHO, M. C. M.; BUZETTI, S.; TEIXEIRA, M. C. M. Estimativa da produtividade de trigo em função da adubação nitrogenada utilizando modelagem neuro fuzzy. **Revista Brasileira de Engenharia Agrícola e Ambiental**, Campina Grande, v. 18, n. 2, p. 180–187, 2014. Disponível em: <<http://www.agriambi.com.br/revista/v18n02/v18n02a08.pdf>>. Acesso em: 12 jan. 2014.
- SILVA, I. A. F.; SILVA, A. A. V.; TEIXEIRA, M. C. M. Modelagem fuzzy com técnicas de agrupamento fuzzy e estruturas neuro fuzzy. In: CONGRESSO BRASILEIRO DE SISTEMA FUZZY-CBSF, 2., 2012, Natal. **Anais...** Natal: SBMAC, 2012. p. 968- 979.
- SILVA, I. N.; SPATTI, S. H.; FLAUZINO, R. A. **Redes neurais artificiais: para engenharia e ciências aplicadas**. São Paulo: Artliber, 2010. 399 p.
- SIVARAMAN, E.; ARULSELVI, S. Modeling of an inverted pendulum based on fuzzy clustering techniques. **Expert Systems with Applications**, New York, v. 38, n. 11, p. 13942-13949, 2011.
- SPECHT, D. F. A generalized regression neural network. **IEEE Transactions on Neural Network**, Piscataway, v. 2, n. 6, p. 569-576, 1991.
- SUGENO, M.; KANG, G. T. Structure identification of fuzzy model. **Fuzzy Sets and Systems**. Amsterdam, v. 28, n. 1, p.15-33, 1988.
- TAKAGI, T.; SUGENO, M. Fuzzy identification of systems and its applications to modeling and control. **IEEE Transactions on Systems, Man and Cybernetics**, New York, v. 15, n. 1, p. 116-132, 1985.
- TANSCHKEIT, R.; GOMIDE, F.; TEIXEIRA, M. C. M. Modelagem e controle nebuloso. In: AGUIRRE, L. A. (Ed.). **Enciclopédia de automática**, São Paulo: Edgard Blucher, v. 3, n. 1, p. 283-324, 2007.

TEIXEIRA, M. C. M.; ASSUNÇÃO, E.; AVELLAR, R. G. On relaxed LMI-based designs for fuzzy regulators and fuzzy observers. **IEEE Transactions on Fuzzy Systems**, New York, v. 11, n. 5, p. 613-623, out. 2003.

TEIXEIRA, M. C. M.; DEAECTO, G. S.; GAINO, R.; CARVALHO, A. A.; FARIAS, U. C. Design of a fuzzy Takagi-Sugeno controller to vary the joint knee angle of paraplegic patients. **Lecture Notes in Computer Science**, Berlin, v. 4234, n. 1, p. 118-126, 2006.

TEIXEIRA, M. C. M.; ZAK, S. H. Stabilizing controller design for uncertain nonlinear systems using fuzzy models. **IEEE Transactions on Fuzzy Systems**, New York, v. 7, n. 2, p. 133-142, abr. 1999.

TEIXEIRA FILHO, M. C. M.; BUZETTI, S.; ANDREOTTI, M.; ARF, O.; BENETT, C. G. S. Doses, fontes e épocas de aplicação de nitrogênio em trigo irrigado em plantio direto. **Pesquisa Agropecuária Brasileira**, Brasília, v. 45, n. 8, p. 797-804, ago 2010.

TEMURTAS, F. A comparative study on thyroid disease diagnosis using neural networks. **Expert Systems with Applications**, Oxford, v. 36, n. 1, p. 944-949, 2009.

UBEYLI, E. D. Combined neural networks for diagnosis of erythematous-squamous diseases. **Expert Systems with Applications**, Oxford, n. 36, n. 3, p. 5107-5112, 2009.

UMPHRED, D. **Reabilitação Neurológica**. 4. ed. Barueri: Manole, 2004. 1200 p.

VANONCINI, M. et al. Electrical stimulation for trunk control in paraplegia: a feasibility study. **Control Engineering Practice**, Oxford, v. 20, n. 12, p. 1247-1258, 2012.

WANG, L. **Adaptive fuzzy systems and control: design and stability analysis**. Englewood Cliffs: Prentice Hall, 1994. 232 p.

WANG, L.; LANGARI, R. Complex systems modeling via fuzzy logic. **IEEE Transactions on Systems, Man, and Cybernetics**, New York, v. 26, n. 1, p. 100-106, 1996.

WASSERMAN, P. D. **Advanced methods in neural computing**. New York: Van Nostrand Reinhold, 1993. p. 155-161.

WERBOS, P. **Beyond regression: new tools for prediction and analysis in the behavioral sciences**. 1974. 906 f. Thesis (Doctor – PHD)- Harvard University, Harvard, 1974.

WRIGHT, I. A.; GOUGH, N. A. J. Artificial neural network analysis of common femoral artery dopple shift signals: classification of proximal disease. **Ultrasound in Med. and Biology**, New York, v. 25, n. 5 p. 735-743, 1999.

WU, D. et al. Prediction of parkinson's disease tremor onset using radial basis function neural networks. **Expert Systems with Applications**, Oxford, v. 37, n. 4, p. 2923-2928, 2010.

YAGER, R. T.; FILEV, D. P. Generation of fuzzy rules by mountain clustering. **Journal of Intelligent and Fuzzy Systems**, Amsterdam, v. 2, n. 3, p. 209-219, 1994.

YAO, Y. Y.; ZHONG, N. Potential applications of granular computing in knowledge discovery and data mining. In: WORLD MULTICONFERENCE ON SYSTEMICS, CYBERNETICS AND INFORMATICS, 1999,. **Proceedings...** [S.l.:s.n.], 1999. p. 573-580.

YOSHIDA, H.; KAWAGUCHI, A.; TSURUYA, K. Radial basis function - sparse partial least squares for application to brain imaging data. **Computacional and Mathematical Methods in Medicine**, New York, , n. UNSP 591032, p. 1-7, 2013.

YU, W.; LI, X. On-line fuzzy modeling via clustering and support vector machines. **Information Sciences**, Berlin, v. 178, n. 22, p. 4264-4279, 2008.

ZADEH, L.A. Fuzzy sets. **Information and Control**. New York, v. 8, n. 3, p. 338-353, 1965.

ZADEH, L. A. Fuzzy algorithms. **Information and Control**. New York, v. 12, n. 2, p. 94-102, 1968.

ZADEH, L. A. Outline of a new approach to the analysis of complex systems and decision processes. **IEEE Transactions on Systems, Man, and Cybernetics**, New York, v.1, , p. 28-44, 1973.

APÊNDICE A

Figura 17 – Medidas dos sensores do pé esquerdo.

PÉ ESQUERDO																							
Sensor	Sensor	Sensor	Sensor	Sensor	Sensor	Sensor	Sensor	Sensor	Sensor	Sensor	Sensor	Sensor	Sensor	Sensor	Sensor	Sensor	Sensor	Sensor	Sensor	Sensor	Sensor	Sensor	Sensor
23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Kg	Kg	Kg	Kg	Kg	Kg	Kg	Kg	Kg	Kg	Kg	Kg	Kg	Kg	Kg	Kg	Kg	Kg	Kg	Kg	Kg	Kg	Kg	Kg
0	0,10067	0	0	2,06376	1,00671	1,15772	1,30872	0,95638	0,15101	0,15101	0,20134	0,05034	0,15101	0,10067	0,10067	0	1,45973	0	2,56711	3,27181	4,47987	0,65436	2,71812
0	0	0,28087	0,18725	0,79579	2,38738	2,15332	1,45115	1,2171	0,79579	1,02985	0,28087	0	0,70217	0,46811	0	0	1,31072	0	1,12347	8,14518	5,7578	0	6,64722
0,04104	0,53355	0,08208	0,04104	2,74982	0,28729	1,35439	0,69771	0,57459	0	1,2723	0,16417	0	0,32834	0,69771	0	0,04104	2,79086	0	2,25731	4,30941	4,51463	0	0
0	0,09197	0,45984	0	1,51747	1,24157	1,74739	1,60944	1,01165	0,32189	1,1496	0,22992	0	0,68976	0,22992	0	0,2759	2,98896	0,13795	4,3225	9,88657	6,75966	1,24157	2,11527
0,04728	0,4728	0	0	1,04016	1,93848	1,56024	3,87696	1,6548	1,32384	0,42552	0,28368	0,09456	0,09456	0,2364	0	0	1,182	0,2364	1,70208	4,53888	4,82256	0,33096	4,11336
0,1887	0,0629	0,1887	0,3774	0,9435	1,1951	1,94989	1,4467	1,3209	0,9435	1,69829	0,6919	0	1,1951	1,0064	0	0,6919	3,14499	0,1258	3,71109	4,59168	3,96269	0,5032	0
0,28653	0,33429	0,38204	0	1,00287	1,05062	2,91309	3,15186	1,52818	0,09551	0,47755	0,09551	0	0,81184	0,04776	0	0,19102	0,76409	0	1,38491	2,149	4,87106	1,19389	2,67431
0	0	0	0	1,14715	0,94765	1,69579	0,94765	1,29678	0,89777	0,59851	0,1995	0	0,54864	0,1995	0,04988	0	1,14715	0,04988	1,59604	2,59356	2,84295	0,64839	3,1422
0	0,31627	0,36898	0	1,26508	1,47593	2,10847	2,68831	2,47746	3,16271	2,16119	2,2139	0,05271	1,00153	2,47746	0,15814	0	2,95186	0,94881	0,47441	3,05729	2,42475	0	0
0	0	0,23041	0	1,26728	0,57604	2,30415	1,26728	2,07373	0	1,26728	0,34562	0	1,84332	0,80645	0	1,95853	5,52995	0	11,6359	3,68664	12,9032	3,34101	0,11521
0,05871	0,23483	0,05871	0	0,35225	1,93738	1,58513	0,82192	0,99804	0,64579	1,46771	0,7045	0	2,23092	0,52838	0	0,41096	2,23092	0,17613	2,17221	4,99022	4,22701	0,11742	4,75538
0,21765	0,32647	0,32647	0,16324	2,33971	1,30588	2,23088	1,57794	1,85	1,46912	1,57794	0,54412	0	2,17647	0,76176	0	0,87059	2,99265	0,10882	3,10147	4,08088	5,76765	4,02647	1,95882
0	0,38462	0,24038	0	0,09615	1,25	1,39423	1,29808	0,86538	0,09615	0,52885	0,14423	0	0,09615	0	0	0,04808	1,44231	0	1,49038	3,75	5,14423	0	2,21154
0,0389	0	0,23342	0,0389	2,21748	1,47832	2,02296	1,71174	1,05039	1,59503	1,12819	0,19452	0,0389	1,08929	0,50574	0	0,11671	1,75064	0,07781	1,01148	2,25638	2,10077	0,0389	2,60651
0	0	0	0	0,30769	1,33333	2,46154	2,66667	1,4359	3,94872	0,46154	0,20513	0	0,46154	0,35897	0	0,15385	1,58974	0,25641	2,61538	2,76923	5,02564	0,97436	0,76923
0	0	0	0,71546	0,91987	0,61325	1,78864	2,29968	2,60631	2,81073	0,35773	0,0511	0	0,0511	0,35773	0,10221	0	2,24858	0	1,99306	3,32177	3,32177	0,20442	1,2776
0,09763	0,09763	0,04881	0	1,56203	1,90373	3,51458	2,39186	1,36678	3,36814	1,12271	0,29288	0,04881	1,0739	0,04881	0	0,14644	1,85492	0	4,73492	4,44203	7,12678	1,22034	1,0739
0,11556	1,15556	0,4237	0,26963	4,66074	1,07852	0,80889	1,00148	5,46963	1,57926	0,15407	0,15407	0	0,6163	0	0	0,11556	0,77037	0	1,81037	3,58222	8,97481	2,6963	5,50815
0,03322	0	0	0,09967	0,1329	1,72764	1,6612	1,29573	1,02994	0,66448	0,66448	0,09967	0	0,6977	0,19934	0,03322	0,53158	2,226	0,09967	3,18949	4,31911	3,72108	1,19606	3,02337
0,10596	0,10596	0,05298	0	1,58938	1,48342	3,01981	2,543	1,58938	1,0066	1,05958	0,10596	0	2,01321	0,05298	0	1,0066	2,17215	0,10596	4,87408	2,43704	4,87408	5,72175	2,96683
0,22098	0,22098	0,16573	0	0,93915	2,26501	2,81745	1,60208	1,16013	0,82866	1,76781	0,22098	0	0,66293	0,22098	0	0,11049	3,81185	0,05524	5,63491	5,08247	5,7454	1,27062	1,65733
0	0,04884	0,29305	0,14652	1,12335	0,53725	0,8303	1,26988	1,46524	1,4164	0,39073	0,14652	0	0,04884	0	0	0,04884	0,5861	0	0,63494	6,00749	5,12834	0,68378	8,35187
0	0,35461	0,20264	0,10132	1,06384	1,62109	2,53295	1,92504	2,33031	2,53295	0,96252	0,15198	0,05066	1,06384	0,20264	0	0,15198	1,31713	0,05066	3,69811	4,05272	3,74876	2,63427	2,07702
0	0	0,04534	0,09067	0	0,45336	0,95205	2,58413	1,72276	1,08806	0,13601	4,98692	0	0,49869	0	0	0,04534	1,72276	0,49869	2,90148	2,13078	3,03749	0,90671	4,03487
0,56454	0,26055	0,56454	0,08685	1,99759	3,17008	7,81665	3,51749	1,60676	0,47768	2,38842	0,13028	0	1,04222	0,1737	0	0	1,99759	0,04343	1,78046	4,95054	3,8649	0,21713	1,95416
0	0,09857	0	0,09857	0,93645	3,64721	1,38003	0,98573	2,02075	0,98573	0,93645	0	0	0,39429	0,24643	0,14786	0,04929	0,83787	0	0,59144	1,87289	2,9572	0,44358	1,92218

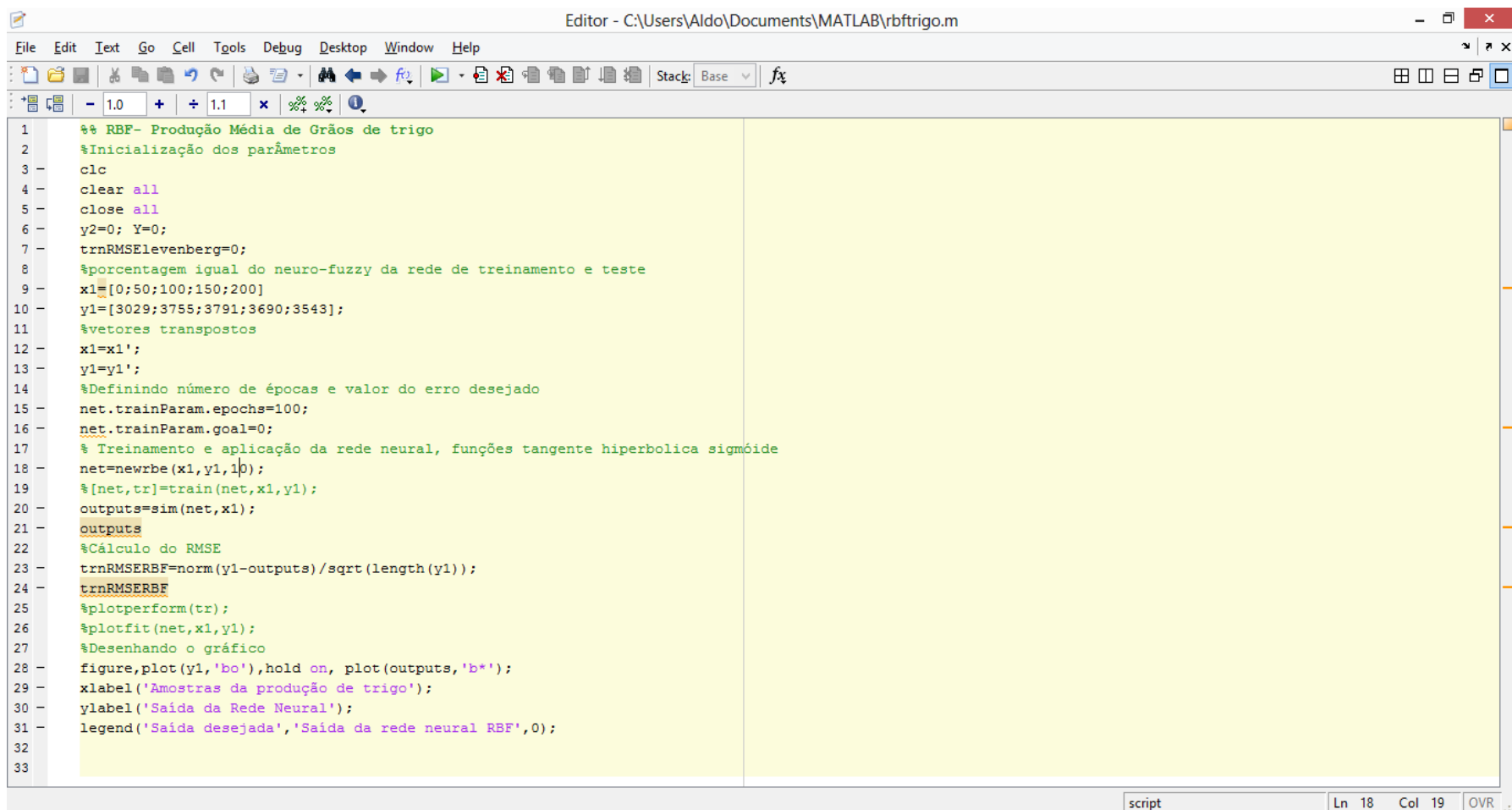
Fonte: Freitas (2008).

Figura 18 – Medidas dos sensores do pé direito.

PÉ DIREITO																							
Sensor	Sensor	Sensor	Sensor	Sensor	Sensor	Sensor	Sensor	Sensor	Sensor	Sensor	Sensor	Sensor	Sensor	Sensor	Sensor	Sensor	Sensor	Sensor	Sensor	Sensor	Sensor	Sensor	Sensor
23	22	21	20	19	18	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
Kg	Kg	Kg	Kg	Kg	Kg	Kg	Kg	Kg	Kg	Kg	Kg	Kg	Kg	Kg	Kg	Kg	Kg	Kg	Kg	Kg	Kg	Kg	Kg
1,10738	0,30201	0	0,65436	0,05034	0	0,55369	4,02685	1,15772	0,7047	0,10067	0	0,45302	0	0,20134	0,35235	0	1,15772	0,40268	2,81879	3,57383	3,97651	0,75503	0
3,74491	0,09362	0	0,09362	0,23406	1,17028	0,93623	1,35753	1,02985	1,49796	0	0,09362	1,12347	0	0,4213	1,17028	0,04681	1,45115	0,56174	6,5536	3,08955	5,47693	3,65129	0,46811
4,30941	0,04104	0,20521	0,36938	0	0	0,69771	1,06709	1,2723	0,45146	0,04104	0,12313	1,76481	0,04104	0,7798	0,61563	0	2,17523	0,32834	2,8319	2,29835	2,66773	0,04104	0,04104
0,22992	0	0	0,18394	0,96567	1,60944	1,28755	3,54077	2,20723	2,2992	0,04598	0	1,19559	0,13795	0,13795	0,78173	0,04598	1,01165	0,68976	4,73636	2,897	5,10423	5,10423	2,75904
1,08744	0,04728	0	0,9456	0,2364	1,08744	1,32384	2,93136	0,99288	2,6004	0,04728	0,18912	0,4728	0,04728	0,14184	0	0,04728	1,22928	0	5,6736	2,8368	6,19368	0,99288	0
0,629	0,5032	0,1887	0,2516	0,1887	1,0693	0,8806	3,77399	1,57249	1,3209	0	0,5032	2,07569	0	0,8177	1,258	0,0629	2,20149	0,629	3,64819	3,01919	3,33369	0,629	0,5032
2,43553	0	0,04776	0,38204	1,05062	0,81184	1,43266	3,19962	1,24164	1,19389	0,19102	0,14327	0,47755	0	0,33429	0,4298	0,04776	0,81184	0,09551	2,96084	0,52531	3,15186	3,53391	0,09551
0,74814	0	0,09975	0,09975	0,09975	1,69579	1,49629	1,09728	0,64839	0,54864	0	0	0,29926	0	0	0,04988	0,04988	0,69827	0	2,59356	0,99752	4,43899	3,89035	0,1995
0,68525	0	0,21085	0,15814	0,21085	2,79373	1,63407	3,16271	1,89763	1,47593	0,05271	1,10695	2,00305	0	1,73949	1,21237	0,21085	2,2139	0,47441	3,26814	2,00305	3,74254	0,15814	0
0,57604	0	0,23041	0,34562	0,11521	0	2,88018	2,76498	1,6129	0,92166	0	0,34562	1,38249	0	0,80645	0,34562	0,11521	4,26267	0,23041	10,5991	3,80184	12,3272	4,7235	0,46083
1,46771	0	0,41096	0,11742	0,05871	0,52838	0,64579	2,17221	1,29159	1,81996	0	0,35225	1,93738	0	0,82192	1,87867	0,11742	2,05479	0,58708	3,81605	2,28963	3,75734	3,17025	0
1,36029	0,10882	0	0,27206	0,16324	0,70735	1,03382	5,06029	1,79559	0,925	0	0,27206	1,79559	0	0,54412	1,46912	0	2,12206	0,97941	4,78824	3,31912	4,40735	4,08088	2,72059
0,19231	0	0	0,19231	0	0,76923	1,29808	2,69231	1,63462	1,20192	0	0,04808	0,52885	0	0	0,09615	0	1,53846	0,04808	1,875	2,25962	5,04808	0,09615	0
2,37309	0,0389	0	0	0	1,12819	0,54464	2,37309	1,82845	1,08929	0,07781	0,07781	1,51722	0,0389	0,35013	1,75064	0	1,2449	1,206	1,32271	1,90626	2,17858	0,77806	0,27232
1,07692	0	0	0	0,05128	3,02564	1,48718	2,61538	1,84615	0,92308	0	0,30769	0,30769	0	0,66667	0,10256	0,05128	1,94872	0	4,35897	2,30769	4,87179	0	0,25641
0,20442	0	1,02208	0,86877	0,25552	0,30662	0,91987	2,04416	1,32871	0,97098	0	0	0,76656	0	0,15331	0,51104	0,0511	1,07319	0,25552	2,453	2,5041	4,39495	2,19748	1,2776
0,68339	0	0,24407	0,09763	0,19525	0,53695	1,51322	6,4922	2,05017	1,0739	0,04881	0	1,85492	0,04881	0,04881	1,51322	0	2,39186	1,70847	1,02508	5,51593	6,05288	0	1,36678
1,6563	0,03852	0,73185	1,81037	0	1,15556	1,50222	7,47259	1,19407	4,50667	0	0	0,46222	0	0,11556	0,53926	0	0,65481	0,26963	3,7363	1,6563	4,69926	3,50519	1,34815
0,09967	0,06645	0	0,09967	0,1329	0,96349	0,99672	1,92699	1,6612	1,42863	0,03322	0,29902	0,63125	0,06645	0,96349	0,03322	1,5283	2,15955	0,03322	3,45529	0,23257	3,95364	3,08982	0,09967
3,97344	0,15894	0	0,05298	0,05298	0,79469	1,32448	1,64235	1,32448	0,74171	0,05298	0	0,68873	0	0,10596	0,42383	0,05298	2,22513	0,21192	5,88069	3,54961	4,98004	5,29792	1,37746
0,71817	0	0	0,11049	0	1,43635	1,16013	3,20416	2,65172	1,32586	0,05524	0,11049	1,04964	0	0,38671	0,11049	0,16573	2,8727	0	5,91113	2,70697	5,69015	2,26501	0,60769
0,14652	0,04884	0,14652	0,14652	0,19537	0,97683	0,87914	2,2467	1,56292	0,97683	0,14652	0	0,5861	0,09768	0,09768	0,29305	0	0,73262	0	3,32121	2,97932	5,51907	2,58859	1,95365
0,20264	0,15198	0	0,30395	0,10132	0,40527	0,96252	2,229	1,36779	2,48229	0,10132	0	0,8612	0,15198	0,15198	0,55725	0,05066	0,40527	0,35461	4,40733	4,50865	4,91392	2,68493	1,31713
0	0	2,08544	0	0,13601	1,85876	1,13339	6,39233	0,09067	1,45074	0	0	0	0	0	0	0	0,72537	0,13601	1,67742	1,72276	4,21622	2,44813	0,09067
2,04101	0,26055	0,52111	0,21713	0,43426	0,56454	1,56333	8,20748	4,29916	2,25814	0,04343	0,13028	2,25814	0,04343	0,39083	0,69481	0,04343	1,56333	0,13028	2,90953	1,47648	2,73583	0,56454	0,04343
0,93645	0,09857	0	0,39429	0,19715	0	0,49287	2,31647	1,4786	0,83787	0,04929	0,14786	0,78859	0	0,49287	0,19715	0,09857	1,33074	0	2,56291	1,33074	3,6965	0	0

Fonte: Freitas (2008).

Figura 22 - Linguagem de script do MATLAB da rede neural RBF para estimativa de produção de trigo ano 2006.



```

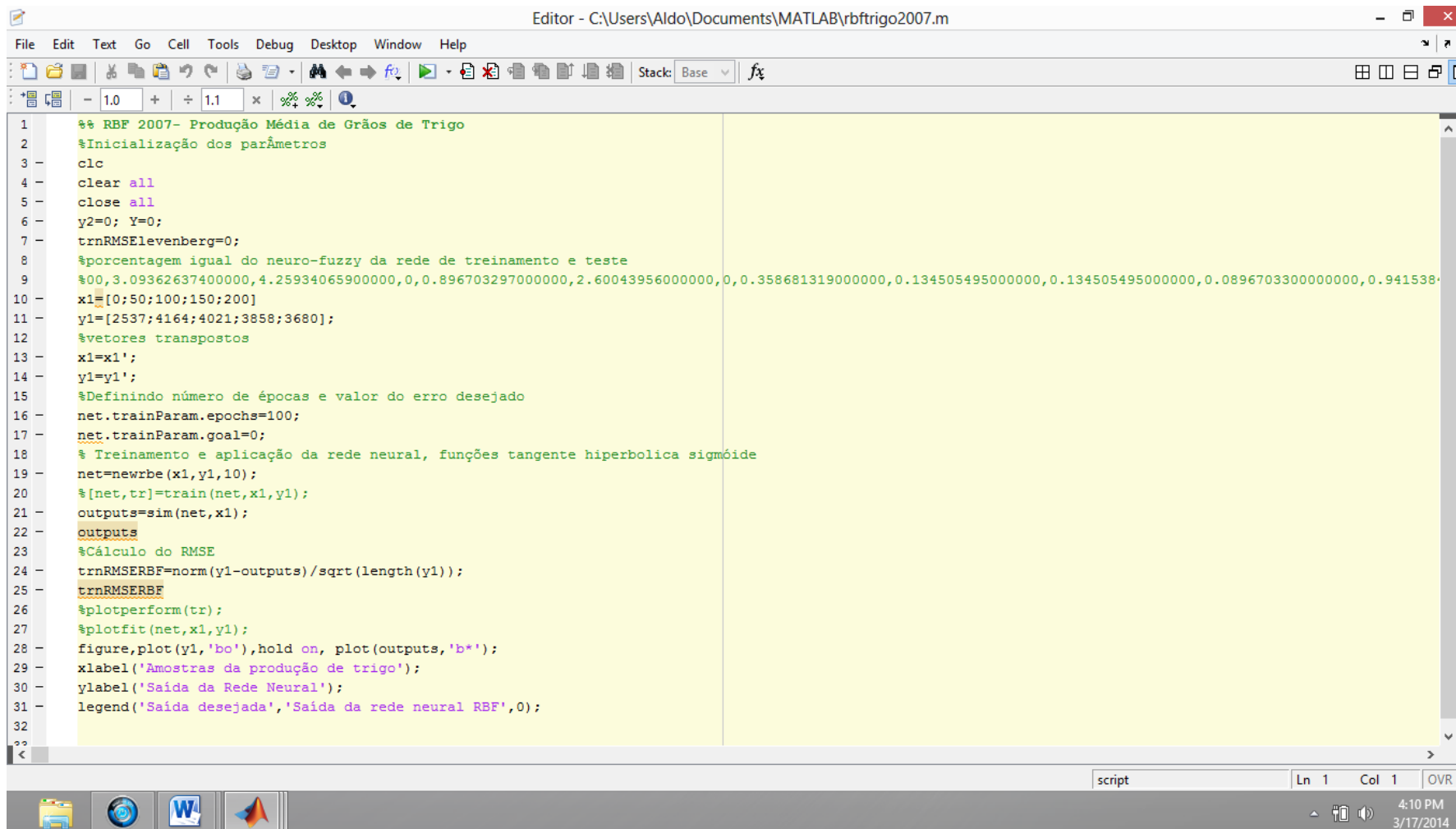
1  %% RBF- Produção Média de Grãos de trigo
2  %Inicialização dos parâmetros
3  clc
4  clear all
5  close all
6  y2=0; Y=0;
7  trnRMSElevenberg=0;
8  %porcentagem igual do neuro-fuzzy da rede de treinamento e teste
9  x1=[0;50;100;150;200]
10 y1=[3029;3755;3791;3690;3543];
11 %vetores transpostos
12 x1=x1';
13 y1=y1';
14 %Definindo número de épocas e valor do erro desejado
15 net.trainParam.epochs=100;
16 net.trainParam.goal=0;
17 % Treinamento e aplicação da rede neural, funções tangente hiperbolica sigmóide
18 net=newrbe(x1,y1,'p');
19 %[net,tr]=train(net,x1,y1);
20 outputs=sim(net,x1);
21 outputs
22 %Cálculo do RMSE
23 trnRMSEBFBF=norm(y1-outputs)/sqrt(length(y1));
24 trnRMSEBFBF
25 %plotperform(tr);
26 %plotfit(net,x1,y1);
27 %Desenhando o gráfico
28 figure,plot(y1,'bo'),hold on, plot(outputs,'b*');
29 xlabel('Amostras da produção de trigo');
30 ylabel('Saída da Rede Neural');
31 legend('Saída desejada','Saída da rede neural RBF',0);
32
33

```

script Ln 18 Col 19 OVR

Fonte: Elaboração do próprio autor.

Figura 23 - Linguagem de script do MATLAB da rede neural RBF para estimativa de produção de trigo ano 2007.



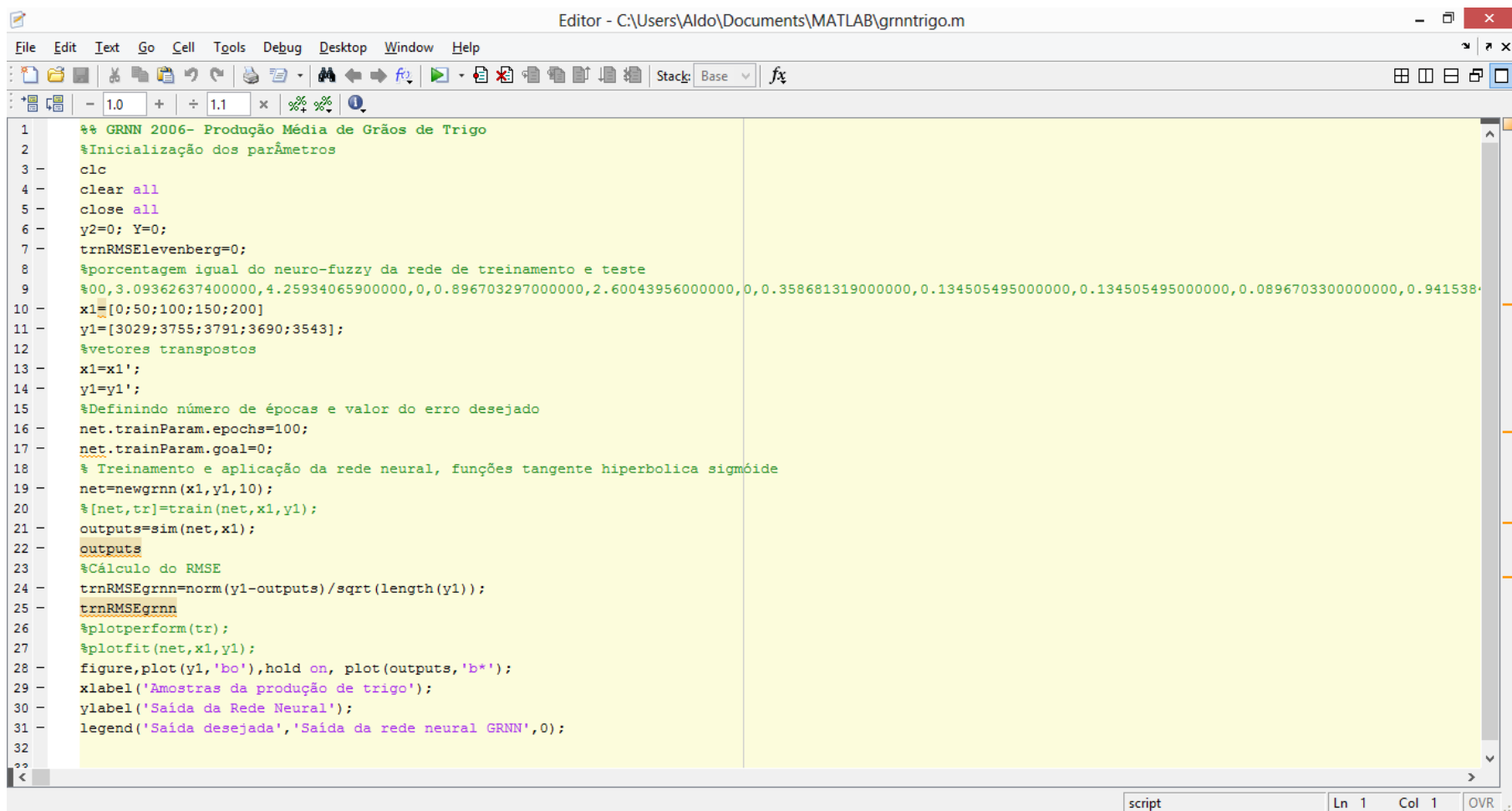
```

1  %% RBF 2007- Produção Média de Grãos de Trigo
2  %Inicialização dos parâmetros
3  clc
4  clear all
5  close all
6  y2=0; Y=0;
7  trnRMSElevenberg=0;
8  %porcentagem igual do neuro-fuzzy da rede de treinamento e teste
9  %00,3.09362637400000,4.25934065900000,0,0.896703297000000,2.60043956000000,0,0.358681319000000,0.134505495000000,0.134505495000000,0.0896703300000000,0.941538
10 x1=[0;50;100;150;200]
11 y1=[2537;4164;4021;3858;3680];
12 %vetores transpostos
13 x1=x1';
14 y1=y1';
15 %Definindo número de épocas e valor do erro desejado
16 net.trainParam.epochs=100;
17 net.trainParam.goal=0;
18 % Treinamento e aplicação da rede neural, funções tangente hiperbolica sigmóide
19 net=newrbe(x1,y1,10);
20 %[net,tr]=train(net,x1,y1);
21 outputs=sim(net,x1);
22 outputs
23 %Cálculo do RMSE
24 trnRMSEERBF=norm(y1-outputs)/sqrt(length(y1));
25 trnRMSEERBF
26 %plotperform(tr);
27 %plotfit(net,x1,y1);
28 figure,plot(y1,'bo'),hold on, plot(outputs,'b*');
29 xlabel('Amostras da produção de trigo');
30 ylabel('Saída da Rede Neural');
31 legend('Saída desejada','Saída da rede neural RBF',0);
32
33

```

Fonte: Elaboração do próprio autor.

Figura 24 - Linguagem de *script* do MATLAB da rede neural GRNN para estimativa de produção de trigo ano 2006.



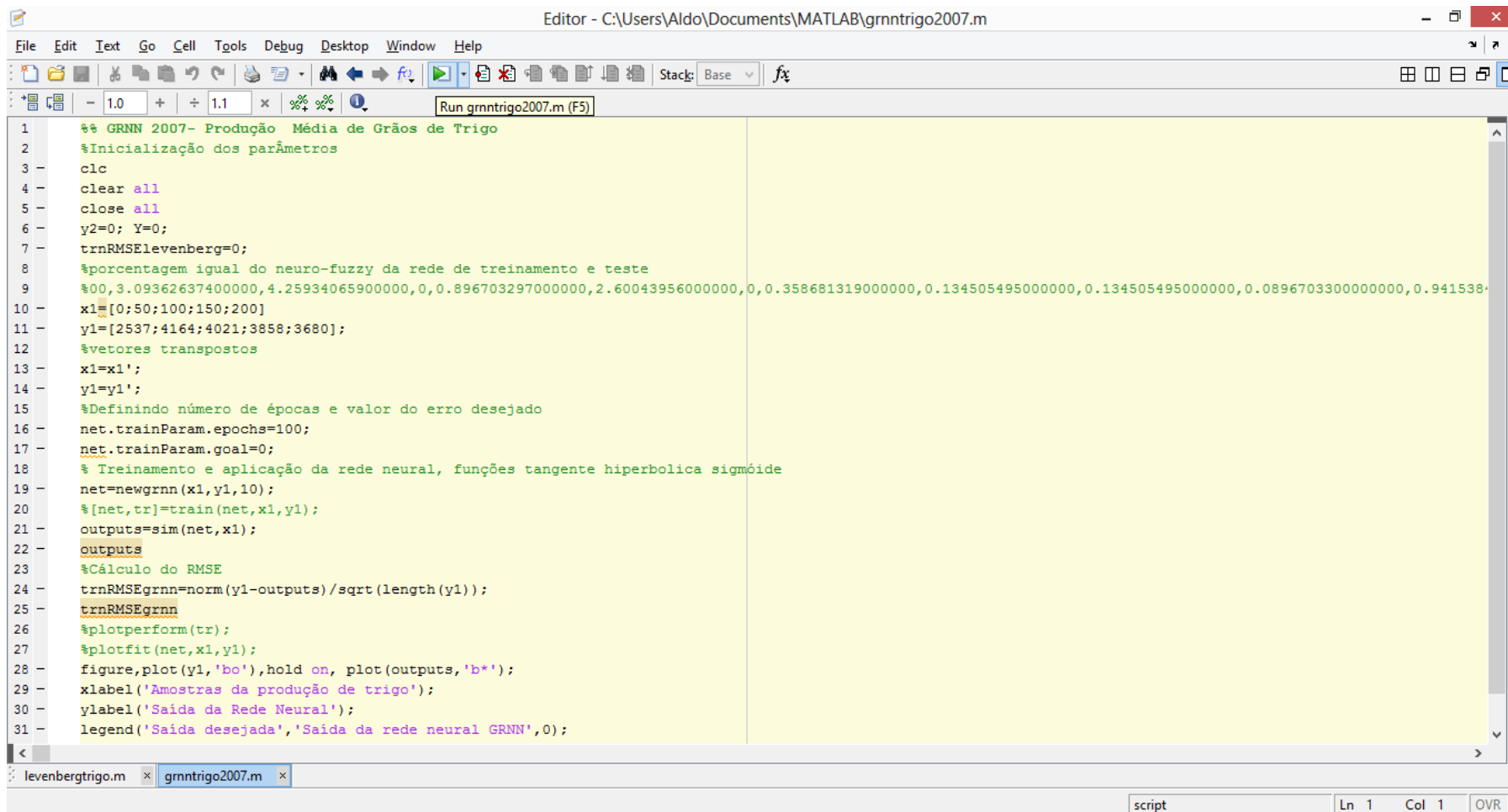
```

1  %% GRNN 2006- Produção Média de Grãos de Trigo
2  %Inicialização dos parâmetros
3  clc
4  clear all
5  close all
6  y2=0; Y=0;
7  trnRMSElevenberg=0;
8  %porcentagem igual do neuro-fuzzy da rede de treinamento e teste
9  %00,3.09362637400000,4.25934065900000,0,0.896703297000000,2.60043956000000,0,0.358681319000000,0.134505495000000,0.134505495000000,0.0896703300000000,0.941538
10 x1=[0;50;100;150;200]
11 y1=[3029;3755;3791;3690;3543];
12 %vetores transpostos
13 x1=x1';
14 y1=y1';
15 %Definindo número de épocas e valor do erro desejado
16 net.trainParam.epochs=100;
17 net.trainParam.goal=0;
18 % Treinamento e aplicação da rede neural, funções tangente hiperbolica sigmóide
19 net=newgrnn(x1,y1,10);
20 %[net,tr]=train(net,x1,y1);
21 outputs=sim(net,x1);
22 outputs
23 %Cálculo do RMSE
24 trnRMSEgrnn=norm(y1-outputs)/sqrt(length(y1));
25 trnRMSEgrnn
26 %plotperform(tr);
27 %plotfit(net,x1,y1);
28 figure,plot(y1,'bo'),hold on, plot(outputs,'b*');
29 xlabel('Amostras da produção de trigo');
30 ylabel('Saída da Rede Neural');
31 legend('Saída desejada','Saída da rede neural GRNN',0);
32
33

```

Fonte: Elaboração do próprio autor.

Figura 25 - Linguagem de script do MATLAB da rede neural GRNN para estimativa de produção de trigo ano 2007.



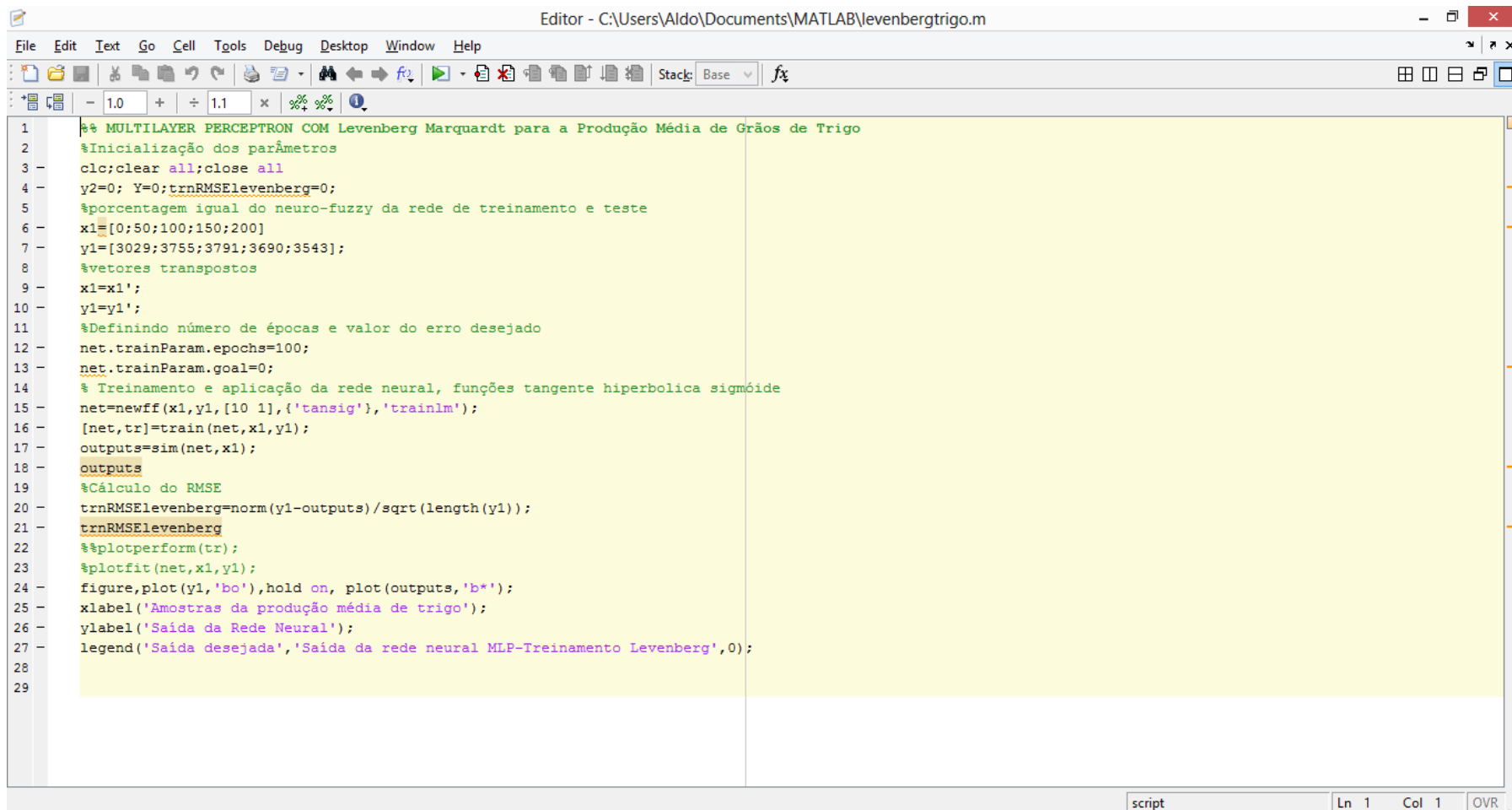
```

1  %% GRNN 2007- Produção Média de Grãos de Trigo
2  %Inicialização dos parâmetros
3  -
4  clear all
5  close all
6  y2=0; Y=0;
7  trnRMSElevenberg=0;
8  %porcentagem igual do neuro-fuzzy da rede de treinamento e teste
9  %00,3.09362637400000,4.25934065900000,0,0.896703297000000,2.60043956000000,0,0.358681319000000,0.134505495000000,0.134505495000000,0.0896703300000000,0.941538
10 - x1=[0;50;100;150;200]
11 - y1=[2537;4164;4021;3858;3680];
12 %vetores transpostos
13 - x1=x1';
14 - y1=y1';
15 %Definindo número de épocas e valor do erro desejado
16 net.trainParam.epochs=100;
17 net.trainParam.goal=0;
18 % Treinamento e aplicação da rede neural, funções tangente hiperbolica sigmóide
19 net=newgrnn(x1,y1,10);
20 %[net,tr]=train(net,x1,y1);
21 outputs=sim(net,x1);
22 outputs
23 %Cálculo do RMSE
24 trnRMSEgrnn=norm(y1-outputs)/sqrt(length(y1));
25 trnRMSEgrnn
26 %plotperform(tr);
27 %plotfit(net,x1,y1);
28 - figure,plot(y1,'bo'),hold on, plot(outputs,'b*');
29 - xlabel('Amostras da produção de trigo');
30 - ylabel('Saída da Rede Neural');
31 - legend('Saída desejada','Saída da rede neural GRNN',0);

```

Fonte: Elaboração do próprio autor.

Figura 26 - Linguagem de script do MATLAB da rede neural MLP para estimativa de produção de trigo ano 2006.



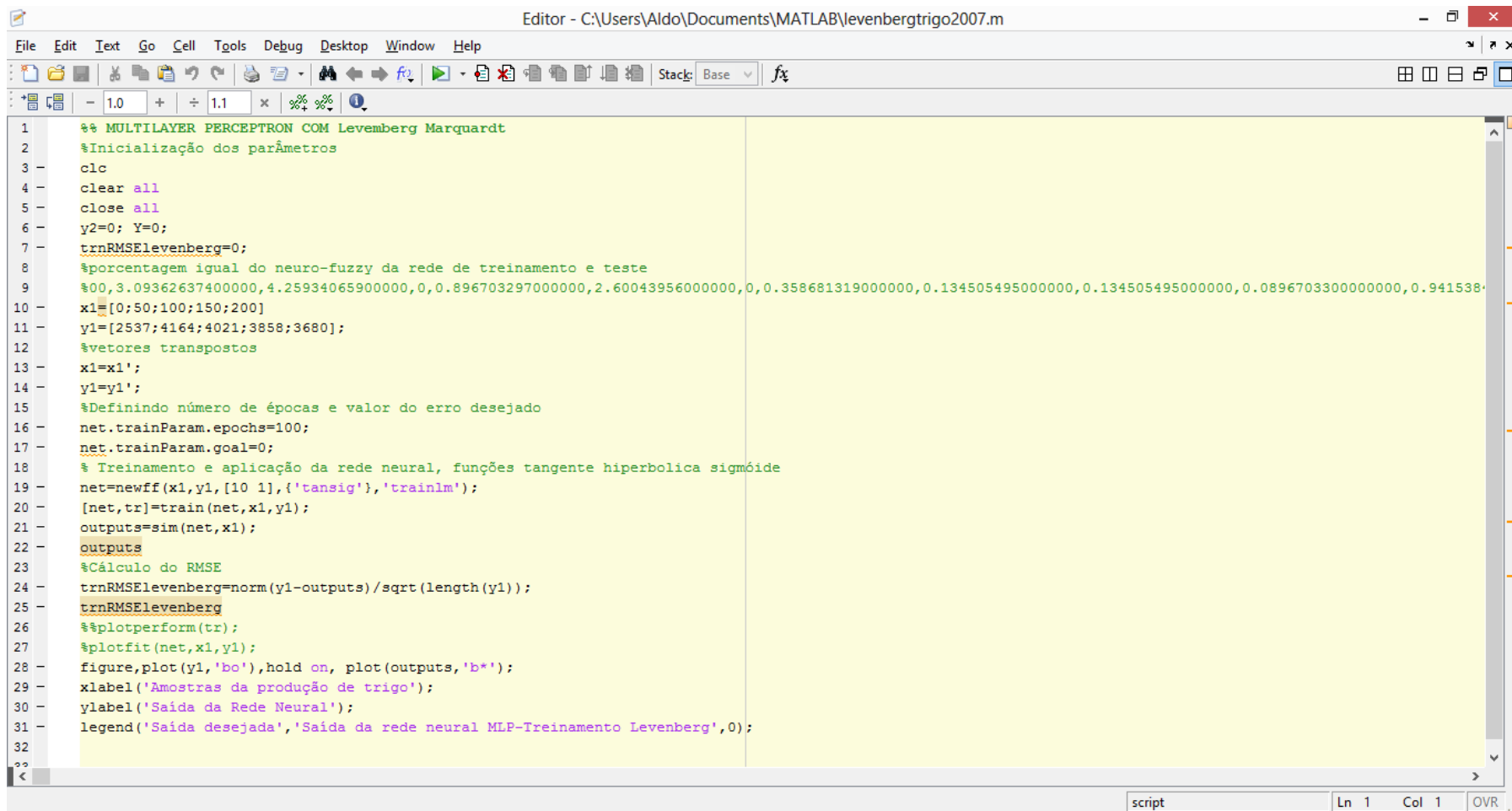
```

1  %% MULTILAYER PERCEPTRON COM Levenberg Marquardt para a Produção Média de Grãos de Trigo
2  %Inicialização dos parâmetros
3  clc;clear all;close all
4  y2=0; Y=0;trnRMSElevenberg=0;
5  %porcentagem igual do neuro-fuzzy da rede de treinamento e teste
6  x1=[0;50;100;150;200]
7  y1=[3029;3755;3791;3690;3543];
8  %vetores transpostos
9  x1=x1';
10 y1=y1';
11 %Definindo número de épocas e valor do erro desejado
12 net.trainParam.epochs=100;
13 net.trainParam.goal=0;
14 % Treinamento e aplicação da rede neural, funções tangente hiperbolica sigmóide
15 net=newff(x1,y1,[10 1],{'tansig','trainlm'});
16 [net,tr]=train(net,x1,y1);
17 outputs=sim(net,x1);
18 outputs
19 %Cálculo do RMSE
20 trnRMSElevenberg=norm(y1-outputs)/sqrt(length(y1));
21 trnRMSElevenberg
22 %%plotperform(tr);
23 %plotfit(net,x1,y1);
24 figure,plot(y1,'bo'),hold on, plot(outputs,'b*');
25 xlabel('Amostras da produção média de trigo');
26 ylabel('Saída da Rede Neural');
27 legend('Saída desejada','Saída da rede neural MLP-Treinamento Levenberg',0);
28
29

```

Fonte: Elaboração do próprio autor.

Figura 27 - Linguagem de script do MATLAB da rede neural MLP para estimativa de produção de trigo ano 2007.



```

1  %% MULTILAYER PERCEPTRON COM Levenberg Marquardt
2  %Inicialização dos parâmetros
3  clc
4  clear all
5  close all
6  y2=0; Y=0;
7  trnRMSElevenberg=0;
8  %porcentagem igual do neuro-fuzzy da rede de treinamento e teste
9  %00,3.093626374000000,4.259340659000000,0,0.896703297000000,2.60043956000000,0,0.358681319000000,0.134505495000000,0.134505495000000,0.0896703300000000,0.941538
10 x1=[0;50;100;150;200]
11 y1=[2537;4164;4021;3858;3680];
12 %vetores transpostos
13 x1=x1';
14 y1=y1';
15 %Definindo número de épocas e valor do erro desejado
16 net.trainParam.epochs=100;
17 net.trainParam.goal=0;
18 % Treinamento e aplicação da rede neural, funções tangente hiperbolica sigmóide
19 net=newff(x1,y1,[10 1],{'tansig'},'trainlm');
20 [net,tr]=train(net,x1,y1);
21 outputs=sim(net,x1);
22 outputs
23 %Cálculo do RMSE
24 trnRMSElevenberg=norm(y1-outputs)/sqrt(length(y1));
25 trnRMSElevenberg
26 %%plotperform(tr);
27 %plotfit(net,x1,y1);
28 figure,plot(y1,'bo'),hold on, plot(outputs,'b*');
29 xlabel('Amostras da produção de trigo');
30 ylabel('Saída da Rede Neural');
31 legend('Saída desejada','Saída da rede neural MLP-Treinamento Levenberg',0);
32

```

Fonte: Elaboração do próprio autor.